Xavier Franch
Tomi Männistö
Silverio Martínez-Fernández (Eds.)

LNCS 11915

Product-Focused Software Process Improvement

20th International Conference, PROFES 2019
Barcelona, Spain, November 27–29, 2019
Proceedings

Profes 2019

Springer

More information about this series at http://www.springer.com/series/7408

Xavier Franch · Tomi Männistö ·
Silverio Martínez-Fernández (Eds.)

Product-Focused Software Process Improvement

20th International Conference, PROFES 2019
Barcelona, Spain, November 27–29, 2019
Proceedings

Springer

Editors
Xavier Franch
Universitat Politècnica de Catalunya
Barcelona, Spain

Tomi Männistö
University of Helsinki
Helsinki, Finland

Silverio Martínez-Fernández
Fraunhofer Institute for Experimental
Software Engineering
Kaiserslautern, Germany

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-030-35332-2 ISBN 978-3-030-35333-9 (eBook)
https://doi.org/10.1007/978-3-030-35333-9

LNCS Sublibrary: SL2 – Programming and Software Engineering

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

On behalf of the PROFES Organizing Committee, we are proud to present the proceedings of the 20th International Conference on Product-Focused Software Process Improvement (PROFES 2019) held in Barcelona. The hosting institution was the Universitat Politècnica de Catalunya - BarcelonaTech in Spain. Since 1999, PROFES has established itself as one of the top recognized international process improvement conferences. In the spirit of the PROFES conference series, the main theme of PROFES 2019 was professional software process improvement (SPI) motivated by product, process, and service quality needs.

PROFES 2019 provided a premier forum for practitioners, researchers, and educators to present and discuss experiences, ideas, innovations, as well as concerns related to professional software development and process improvement driven by product and service quality needs. At PROFES 2019, solutions found in practice and relevant research results from academia were presented.

A committee of leading experts in software process improvement, software process modeling, and empirical software engineering selected the technical program. This year, 65 full research papers were submitted. At least three independent experts reviewed each paper. After thorough evaluation, 24 technical full papers were finally selected (37% acceptance rate). In addition, four out of nine industrial papers were selected to the program.

Furthermore, we received 30 short paper submissions. Each submission was reviewed by three members from the PROFES Program Committee. Based on the reviews and overall assessments, 11 short papers were accepted for presentation at the conference and for inclusion in the proceedings (37% acceptance ratio).

Continuing the open science policy in the previous PROFES 2017 and PROFES 2018, we encouraged and supported the authors of all accepted submissions to make their papers and research publicly available.

The topics addressed in this year's papers indicate that SPI is still a vibrant research discipline, but is also of high interest for industry. Several papers report on case studies or SPI-related experience gained in industry. The accepted papers of PROFES 2019 addressed, for example, the following topics:

– Continuous Delivery and Experimentation
– Software Testing
– Software Development
– Technical Debt
– Estimations
– Microservices

Since the beginning of the PROFES conference series, the purpose has been to highlight the most recent findings and novel results in the area of process improvement. We were proud to have Professor Neil Maiden (City, University of London) and

Jennifer Nerlich (Vogella), two renowned keynote speakers from research and industry, at the 2019 edition of PROFES.

Further relevant topics were added by the events co-located with PROFES 2019: the Third International Workshop on Managing Quality in Agile and Rapid Software Development Processes, the 4th International Workshop on Human Factors in Software Processes, and four tutorials addressing themes relevant to industry. The role of two European space co-chairs was added to the Organizing Committee. Responsibilities included providing an opportunity for researchers involved in ongoing and/or recently completed research projects (national, European, and international) related to the topics of the conference to present their projects and disseminate the objectives, deliverables, or outcome. Complementing the main scientific program, these events were included in the program to bring together researchers and representatives from industry by providing researchers with the opportunity to attend industry tutorials and providing practitioners with the latest research.

We are thankful for the opportunity to have served as chairs for this conference. The Program Committee members and reviewers provided excellent support in reviewing the papers. We are also grateful to all authors of submitted manuscripts, presenters, and session chairs for their time and effort in making PROFES 2019 a success. We would also like to thank the PROFES Steering Committee members for the guidance and support in the organization process. Furthermore, we thank everyone in the organization team as well as the student volunteers for making PROFES 2019 an experience that will live on in the participants' memory for years to come.

November 2019

Xavier Franch
Tomi Männistö
Silverio Martínez-Fernández

Organization

Organizing Committee

General Chair

Xavier Franch	Technical University of Catalunya, Spain

Program Co-chairs

Tomi Männistö	University of Helsinki, Finland
Silverio Martínez-Fernández	Fraunhofer IESE, Germany

Short Paper Co-chairs

Oscar Dieste	Universidad Politécnica de Madrid, Spain
Pilar Rodríguez	University of Oulu, Finland

Industry Paper Chair

Danilo Caivano	SER & Practices, Italy

Workshop Co-chairs

Casper Lassenius	Aalto University, Finland
Andreas Vogelsang	Technical University of Berlin, Germany

Tutorial Co-chairs

Matthias Galster	University of Canterbury, New Zealand
Dietmar Pfahl	University of Tartu, Estonia

Journal-First Track Chair

Daniel Méndez Fernández	Blekinge Institute of Technology, Sweden, and fortiss GmbH, Germany

European Project Space Co-chairs

Alessandra Bagnato	Softeam, France
Davide Fucci	HITeC – University of Hamburg, Germany

Organization Chair

Carme Quer	Technical University of Catalunya, Spain

Proceedings Co-chairs

Claudia Ayala	Technical University of Catalunya, Spain
Jordi Marco	Technical University of Catalunya, Spain

Social Media and Publicity Co-chairs

Marc Oriol	Technical University of Catalunya, Spain
Anna Maria Vollmer	Fraunhofer IESE, Germany

Webmaster

Carles Farré	Technical University of Catalunya, Spain
Maria José Salamea	Technical University of Catalunya, Spain

Contact Person

Dolors Costal	Technical University of Catalunya, Spain

Local Organization Team

Katarzyna Biesialska	Technical University of Catalunya, Spain
Xavier Burgués	Technical University of Catalunya, Spain
Cristina Gómez	Technical University of Catalunya, Spain
Lidia López	Technical University of Catalunya, Spain
Martí Manzano	Technical University of Catalunya, Spain
Cristina Palomares	Technical University of Catalunya, Spain

Program Committee

Full Research and Industry Papers Program Committee

Silvia Abrahao	Universitat Politècnica de València, Spain
Sousuke Amasaki	Okayama Prefectural University, Japan
Maria Teresa Baldassarre	University of Bari, Italy
Vita Santa Barletta	University of Bari, Italy
Stefan Biffl	Vienna University of Technology, Austria
Andreas Birk	SWPM, Germany
Luigi Buglione	Engineering Technology Services (ETS), Italy
Gerardo Canfora	University of Sannio, Italy
Bruno da Silva	California Polytechnic State University, USA
Maya Daneva	University of Twente, The Netherlands
Michal Dolezel	University of Economics – Prague, Czech Republic
Christof Ebert	Vector, Germany
Fabian Fagerholm	University of Helsinki, Finland
Davide Falessi	California Polytechnic State University, USA
Masud Fazal-Baqaie	Fraunhofer, Germany
Michael Felderer	University of Innsbruck, Austria
Davide Fucci	HITeC – University of Hamburg, Germany
Lina Garcés	University of São Paulo, Brazil

Carmine Gravino	University of Salerno, Italy
Daniel Graziotin	University of Stuttgart, Germany
Noriko Hanakawa	Hannan University, Japan
Frank Houdek	Daimler AG, Germany
Andrea Janes	Free University of Bolzano, Italy
Petar Jovanovic	Universitat Politècnica De Catalunya – Barcelona Tech, Spain
Oliver Karras	Leibniz Universität Hannover, Germany
Petri Kettunen	University of Helsinki, Finland
Jil Klünder	Leibniz Universität Hannover, Germany
Jingyue Li	Norwegian University of Science and Technology, Norway
Lidia López	Universitat Politècnica De Catalunya – Barcelona Tech, Spain
Stephen MacDonell	University of Otago, New Zealand
Kenichi Matsumoto	Nara Institute of Science and Technology (NAIST), Japan
Maurizio Morisio	Politecnico di Torino, Italy
Maleknaz Nayebi	Ecole Polytechnique de Montréal, Canada
Risto Nevalainen	Spinet Oy, Finland
Edson Oliveira Jr.	State University of Maringá, Brazil
Paolo Panaroni	INTECS, Italy
Dietmar Pfahl	University of Tartu, Estonia
Rudolf Ramler	Software Competence Center Hagenberg, Austria
Daniel Rodriguez	The University of Alcalá, Spain
Simone Romano	University of Basilicata, Italy
Bruno Rossi	Masaryk University, Czech Republic
Gleison Santos	Federal University of the State of Rio de Janeiro, Brazil
Giuseppe Scanniello	University of Basilicata, Italy
Klaus Schmid	University of Hildesheim, Germany
Kari Smolander	Lappeenranta University of Technology, Finland
Martin Solari	Universidad ORT, Uruguay
Michael Stupperich	Daimler AG, Germany
Guilherme Travassos	Federal University of Rio de Janeiro, Brazil
Rini Van Solingen	Delft University of Technology, The Netherlands
Antonio Vetrò	Politecnico di Torino, Italy
Stefan Wagner	University of Stuttgart, Germany
Hironori Washizaki	Waseda University, Japan
Dietmar Winkler	Vienna University of Technology, Austria

Short Papers Program Committee

Muhammad Ovais Ahmad	Karlstad University, Sweden
Elina Annanperä	University of Oulu, Finland
Beatriz Bernárdez Jiménez	Universidad de Sevilla, Spain
Dante Carrizo	University of Atacama, Chile

Jessica Díaz	Universidad Politécnica de Madrid, Spain
Efraín R. Fonseca C.	Universidad de las Fuerzas Armadas ESPE, Ecuador
Davide Fucci	HITeC — University of Hamburg, Germany
Vahid Garousi	Queen's University Belfast, UK
Itir Karac	University of Oulu, Finland
Kati Kuusinen	Technical University of Denmark, Denmark
Lucy Ellen Lwakatare	Chalmers University of Technology, Sweden
Marc Oriol	Universitat Politècnica de Catalunya, Spain
Simone Romano	University of Basilicata, Italy
Norsaremah Salleh	International Islamic University Malaysia, Malaysia
Davide Taibi	Tampere University of Technology, Finland
Xiaofeng Wang	Free University of Bozen–Bolzano, Italy

Additional Reviewers

Corrado Aaron Visaggio
Monica Anastassiu
Justus Bogner
Eliezer Dutra
Jonas Fritzsch
Hong Guo
Nektaria Kaloudi
Vasileios Theodorou
Eugenio Zimeo

Intertwining Creative and Design Thinking Processes for Software Products (Keynote Abstract)

Neil Maiden

Cass Business School, City, University of London, 106 Bunhill Row, London EC1Y 8TZ, UK
N.A.M.Maiden@city.ac.uk

Abstract. Most software development processes still pay little attention to creativity and creative thinking, even though creative outcomes are pre-requisites for innovation. The recent interest in design thinking methods places shifts the focus to both software products and processes, but still does not address the creativity deficit of most design thinking practices. This keynote presentation and paper proposes an alternative and more effective framing of design thinking – as situated uses of creativity techniques and design artefacts, opportunistically, in agile development processes. It will introduce the role of design thinking as creative thinking for specific ends. It will summarize common characteristics of high-performance design behaviours – behaviours that are often impeded by software development methods. It will then demonstrate, with multiple examples, how coupling creativity techniques with playful artefacts for design thinking can lead to original design outcomes, often more productively, than with existing software development processes and models.

Keywords: Software development · Software product · Creativity

1 Creativity, Design Thinking and Innovation

Creativity and creative thinking have emerged as essential capabilities of most businesses. It has become a strategic, macro-economic activity, replacing the focus on information at the end of the last century. The World Economic Forum identified it to be a top-three need for economic growth in the next decade, alongside complex problem solving and critical thinking. It is identified as a precondition for business success – for example an IBM survey of 1500 CEOs identified creativity as the leading need and differentiator in their businesses [3]. It is also recognized as a critical pre-condition to effective innovation, generating new forms of creative capitalism based on knowledge and talent. And as digital technologies have become critical to the functioning of many organizations, creativity assumes a more important role in the specification and design of these technologies. Unfortunately, few methods and techniques for software product development explicitly support creative thinking by developers or stakeholders.

Outside of software product development, creative thinking is core to early design activities. For example, the United Kingdom's Design Council defines design as shaping ideas to become practical and attractive propositions for users or customers, and it can be described as creativity deployed to a specific end. Design is both a creative and user-centred approach to problem solving that cuts across different professions, from art and design to engineering and architecture. As such, creativity is needed to generate new ideas that design can shape to become the practical and attractive propositions for users or customers [2].

To deliver more creative design processes over the last decade, design thinking has become accepted practice for many forms of product and service. Design thinking is a human-centred innovation process that involves observation, collaboration, fast learning, the visualization of ideas and rapid prototyping, all of which run concurrent to business analysis activities [4]. It has been successfully used in projects to design new workplaces, consumer products and even brands.

However, one criticism that can be leveled at most design thinking processes is the lack of explicit use of creativity techniques from creative problem solving communities. Indeed, we observe an increasing disconnect between design thinking and creative problem solving, and believe that new techniques and tools that bridge the outputs of these communities are needed. More connected creative problem solving and design thinking methods and techniques can impact on the development of many forms of service and product, including software products.

This keynote proposes an alternative and more effective framing of design thinking – as situated uses of creativity techniques and design artefacts, opportunistically, in agile and other software development processes. It will introduce the role of design thinking as creative thinking for specific ends. It will summarize common characteristics of high-performance design behaviours – behaviours that are often impeded by software development methods. It will then demonstrate, with multiple examples, how coupling creativity techniques such as constraint removal [5] and creativity triggers [1] with playful artefacts for design thinking such as storyboards and desktop walkthroughs [6] can lead to original design outcomes, often more productively, than with existing software development processes.

References

1. Burnay, C., Horkoff, J., Maiden, N.: Stimulating stakeholders' imagination: new creativity triggers for eliciting novel requirements. In: Proceedings of IEEE International Requirements Engineering Conference, 12–16 September 2016, Beijing, China (2016)
2. Design Council, Design for Innovation (2011). https://www.designcouncil.org.uk/sites/default/files/asset/document/DesignForInnovation_Dec2011.pdf
3. IBM, IBM Global CEO Study: Capitalizing on Complexity (2010)
4. Lockwood, T.: Design Thinking, Allworth Press, New York (2010)
5. Maiden, N.A.M., Robertson, S.: Integrated creativity into requirements processes: experiences with an air traffic management system. In: Proceedings of 13th IEEE International Conference on Requirements Engineering, 105–114. IEEE Computer Society Press (2015)
6. Stickdorn, M., Schneider, J.: This is Service Design Thinking, BIS Publishers (2010)

Contents

Continuous Delivery

Agile

Project Management

Microservices

Continuous Experimentation

European Project Space

4th International Workshop on Human Factors in Software Development Processes (HuFo)

Short Tutorials

Evaluating the Utility of the Usability Model for Software Development Process and Practice

Diego Fontdevila[1(✉)], Marcela Genero[2], Alejandro Oliveros[1], and Nicolás Paez[1]

[1] Universidad Nacional de Tres de Febrero, Caseros, Argentina
{dfontdevila, aoliveros}@untref.edu.ar, nicopaez@computer.org

[2] Department of Technologies and Information Systems, University of Castilla-La Mancha, Ciudad Real, Spain
marcela.genero@uclm.es

Abstract. Processes and practices are tools that organizations use to improve their capabilities. Agile transformations are very popular, as are process and practice improvement and adoption initiatives, but they face many challenges, including low adoption rates. Improving process and practice usability might increase adoption rates and effective use. This idea led us to define a Usability Model for Software development Process and Practice (UMP), consisting of characteristics and metrics, in the quest to improve the work experience of software development practitioners and the effectiveness of process and practice adoption initiatives. The goal of this paper is two-fold: (1) to present the refined version of the UMP and (2) to describe a study on the application of the UMP to the Visual Milestone Planning (VMP) method in order to evaluate UMP's utility, specifically its ability to produce useful feedback in a real-life scenario. The study produced preliminary confirmation that the UMP is applicable to the VMP, along with specific feedback on improvement opportunities for the VMP. An interview with the VMP creator confirmed that the UMP model and the evaluation feedback were valuable for enhancing VMP adoption. In summary, we can conclude that the empirical results obtained show that UMP can be useful. Nonetheless, more studies are needed to provide further confirmation in different scenarios.

Keywords: Usability · Process and practice · Improvement · Interview · Design Science Research

1 Introduction

Improvement in software engineering increasingly takes the form of initiatives to adopt pre-existing processes, practices, methods, frameworks, etc. As used to be the case with generic "best practices", agile methods have become a focus of popular interest in the software development community (and beyond), and many organizations are attempting agile and digital transformations [1]. Agile methods, like other related trends such as DevOps, tend to be very attractive to newcomers, but sometimes seem deceptively simple and easy to implement [2]. Research on actual agile projects shows

X. Franch et al. (Eds.): PROFES 2019, LNCS 11915, pp. 741–757, 2019.
https://doi.org/10.1007/978-3-030-35333-9_57

significant differences between method definitions and actual implementations [3], including lack of implementation of many of the practices defined by those same methods [4, 5]. This affects the ability of these initiatives to produce their intended results. For example, agile engineering practices like Frequent Delivery have been shown to be key success factors in agile projects [6, 7].

There is also growing evidence that human factors, emotions in particular, affect software development productivity, turnover and job satisfaction [8]. Furthermore, developer acceptance of new ways of working is a cornerstone of success for improvement initiatives [9, 10]. Overall, the quality of interactions between practitioners and their processes and practices affects the chances of success of improvement initiatives and their effectiveness.

Although there are several process quality models [11, 12], none of them focuses specifically on process usability. In its effort to improve both the overall experience of practitioners and their effectiveness, the UMP provides support for the improvement of the design of processes and practices, as well as their adoption plans. In particular, the UMP should help practitioners to:

- Have a better understanding of usability issues related to processes and practices.
- Evaluate the fitness of potential processes or practices to specific contexts (for example, mature teams might be better suited to hard-to-learn but potentially beneficial practices).
- Adapt processes or practices by highlighting specific concerns (e.g. particular characteristics), in an attempt to enhance the adoption process.
- Support planning of improvement initiatives by providing specifics on usability related risks.
- Provide explanation for obstacles found in the adoption process.

In this context we have defined the long-term goal of our research as:

> Define and validate a usability model for software development process and practice (UMP) to support the enhancement of usability aspects of process and practice, in order to improve the work experience of software development practitioners and the overall effectiveness of process and practice adoption initiatives.

To achieve this main goal, a Design Science approach was followed. Design Science proposes creating artifacts to solve a practical problem, together with knowledge that is of general interest [13]. The research methodology shown in Fig. 1 was followed. For each step the figure shows the associated Design Science activity [13] and the research sub-activities performed:

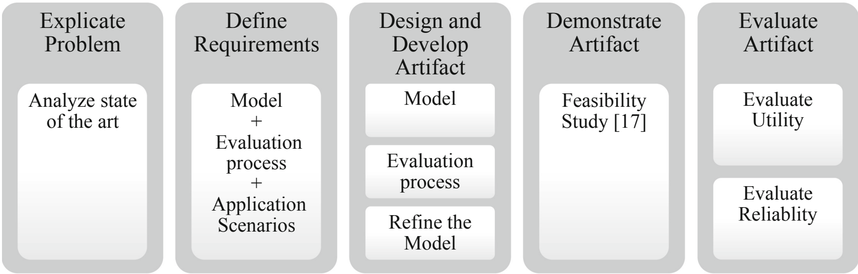

Fig. 1. Design Science Research methodology overview

To allow readers to understand the complete investigation process to which the current paper belongs, the main aspects of the research methodology are introduced briefly as follows:

Explicate Problem: The initial analysis of the state of the art was conducted, in which the relevant literature was identified. Expert interviews were conducted to help identify appropriate sources for the model, given the very limited search results that matched process and practice usability research.

Define Requirements: The artifacts to be produced would be a quality model and an evaluation process. Moreover, real life UMP application scenarios were defined (see Appendix) to describe in detail the intended scope of applicability, including who would use the model, what they would use it for, and which model elements (e.g. empty model or model evaluation results) would be used.

Design and Develop Artifact: The UMP model was defined from the following sources: the ISO 25010 international standard [14], the process quality model by Kroeger et al. [11], and classic usability literature [15, 16].

To facilitate the application of the UMP, an evaluation process was defined by adapting the reference model for software product evaluation proposed in the ISO 25040 international standard [17].

To refine the model, a focus group [19] with expert practitioners was conducted to obtain feedback related to the clarity, understandability, precision, and relevance of model characteristics and metrics (see Appendix for the detailed data collected). Finally, the UMP was modified to address the improvement opportunities identified in the focus group.

Demonstrate Artifact: An initial feasibility study was conducted by applying the model to Scrum (this was the initial publication of the model [18]). Although Fig. 1 might seem like a sequence, the Design Science method framework is iterative; as an example, the feasibility study [18] was performed before the refinement presented here.

Evaluate Artifact: A preliminary study was conducted through the evaluation of the Visual Milestone Planning (VMP) method [20]. This was at the request of the method creator, who was interested in an external evaluation. The objective of this study was to evaluate UMP utility, i.e. its ability to produce actionable feedback that can be used in real-life scenarios to improve the adoption process for specific processes and practices. Other studies will be conducted to complement this preliminary study, in order to provide stronger evaluation.

UMP reliability will be evaluated by asking practitioners to fill in a survey on applying the model to specific processes and practices, and by assessing the inter-evaluator agreement on the evaluations.

The goal of the current paper is two-fold: (1) to present the refined version of the UMP (produced during the Design and Develop Artifact activity), and (2) to describe a study on the application of the model to the Visual Milestone Planning (VMP) method to evaluate UMP's utility, i.e. its ability to produce actionable feedback that is useful in a real-life scenario (first step of the Evaluate Artifact activity).

The rest of this paper is structured as follows: Sect. 2 presents work related to Process and Practice Usability; Sect. 3 describes the UMP model in detail, including its characteristics and metrics; Sect. 4 provides a description of the study on applying UMP to the evaluation of VMP; Sect. 5 reviews the threats to validity, and Sect. 6 outlines the conclusions and future lines of work.

2 Related Work

In this section, we present literature related to process and practice usability:

- Feiler and Humphrey describe the challenges of improving process usability due to long feedback loops, but do not include it in their list of process quality attributes [12].
- Culver-Lozo limits the analysis to process documentation usability [21].
- Cockburn has reflected on the concept of high-discipline methodologies, which he describes as those hard to sustain, and requiring a specific mechanism to keep them in place [22]. This distinction touches on one aspect of the relationship between methodologies and their users, through the associated risk of abandonment.
- Riemenschnaider et al. have found that practitioner acceptance of methodologies can be strongly influenced by subjective norm, i.e., acceptance by close members of the same organization. This highlights the importance of context and the social aspect of usability beyond individual interactions [9].
- Kroeger et al. [11] built their process quality model through a sound grounded theory research. The emergent process quality attributes were organized into 4 groups: Suitability, Usability, Manageability and Evolvability. Usability emerged as a grouping of: Learnability, Understandability, Accessibility and Adaptability. Though their process quality model emerged from interviews with practitioners, its sub-attributes have little relationship with actual process performance by users.
- The ISO 25010 Standard on Systems and software quality models is a product-oriented international standard that includes usability aspects. The process as software analogy [12] supports the inclusion of software usability, given there is no software development process quality standard. This standard was the only source that provided specific metrics for the UMP.

- The works of Norman [15] and Nielsen [16] provided deep product usability concepts and rich terminology.

To our knowledge, research on process and practice usability is very limited, since most existing work does not consider people as users of their processes and practices. We propose the UMP as a means of filling this gap, to help improve the experience of practitioners and the overall effectiveness of adoption initiatives.

3 The UMP Usability Model for Process and Practice

We defined the UMP [18] to help consultants, researchers, teachers and practitioners to enhance the usability aspects of software development processes and practices in order to improve the adoption experience for newcomers and practitioners of software development processes, practices and methods. The UMP consists of several artifacts: The UMP itself (characteristics and metrics definitions), the UMP Evaluation Process, and the Usability Profile (metric values and evaluation comments with improvement recommendations) resulting from the evaluation of a specific process and practice.

The UMP can be used in several modes:

- **Evaluation:** the UMP is used to evaluate a specific process or practice and thus produce a usability profile with improvement recommendations. In this mode, the goal of the model user is to get systematic feedback on the process/practice under evaluation. The UMP itself and the evaluation process are used to produce the usability profile with improvement recommendations.
- **Profile:** the UMP was previously used by a third-party to perform an evaluation and now the user applies the results of that evaluation to a specific context (e.g. team considering adopting a specific practice, as in Scenario #4, see Appendix). In this mode, the usability profile is the only artifact used.
- **Framework:** the UMP is used as a usability framework for process and practice improvement, acting as a checklist that provides potential risks/root causes that can assist in planning and assessing adoption/improvement initiatives. In this mode it also provides metrics that can be used to assess the improvement initiative.

Given that the model is rather complex (its 10 characteristics aimed at being complete), and that it tends to require significant experience with process for someone to be able to perform effective evaluations, these modes allow practitioners to eventually benefit from third-party (and even reusable) expert evaluation results (in the Profile mode) or to use only parts of the model in the Framework mode.

This section presents the refined version of the UMP. The model characteristics modified from the original version of the model [18] during the refinement process are marked with an asterisk in Table 1.

The construction of the UMP consisted of defining the 10 usability characteristics and 24 metrics that the model is composed of. The construction process was based on an adaptation of the top-down methodology for building structured quality models [23], which proposes starting with the top-level elements (i.e. characteristics) and proceeding to the lower level elements (i.e. metrics). More details on the model development process are available in [18].

Table 1 presents the model characteristics, which apply to several aspects of the process and practice adoption lifecycle. For example, for process and practice adoption planning: Self-evident Purpose, Understandability, Learnability, Attractiveness; for process and practice performance: Visibility, because it characterizes how transparent the status and intermediate products of a process are to its stakeholders; Controllability, because it describes how easy it is for different stakeholders to control a process or practice during execution; and User satisfaction, which is a by-product of the experience of using the process or practice. This does not mean that other characteristics might not support those activities as well, but it highlights the fact that in different contexts different sets of characteristics might prove more significant.

Table 1. UMP characteristics

Characteristic	Definition
Self-evident purpose*	Ease with which users can recognize what a process or practice is for by its name
Learnability*	Ease with which a process or practice user is able to learn how to perform its activities at a novice level of ability [24]
Understandability*	Ease with which a process or practice user is able to apprehend how the underlying principles, structure and dynamics make it work to achieve the desired results
Safety*	Degree to which a process or practice is safe for its users, preventing errors, including using the practice or process incorrectly, or limiting the impact of such errors
Feedback*	Degree to which use of a process or practice produces or promotes reactions or responses to actions performed
Visibility*	Degree to which a process or practice helps make activities, status, obstacles and information inputs and outputs visible to people
Controllability*	Degree to which a process or practice allows its users to check status and make decisions that affect the outcomes during process or practice execution
Adaptability	Ease with which a process or practice user is able to adapt the process or practice for use in different contexts
Attractiveness	Degree to which users of the process or practice find it attractive or appealing because of its form, structure or reported results
User satisfaction	Degree to which user needs are satisfied when using a process or practice

The Goal Question Metric (GQM) [25, 26] paradigm was used for the definition of the metrics for each characteristic. For each metric, several meta-data were defined: description, measurement method, type of scale (e.g. nominal), scale (e.g. yes/no), unit of measurement and most favorable value. The meta-data fields were selected based on the ISO 15939 Systems and Software Engineering – Measurement process Standard

[27]. Care was taken to keep the model as simple as possible and to improve ease of use of the metrics. Overall, metrics were changed significantly during model refinement and were simplified to enhance the experience of model users, based on feedback from the focus group (see Appendix for details).

Table 2 shows the definition of the metrics for each characteristic:

Table 2. Overview of UMP metrics for each characteristic

Characteristic	Metric	Definition	Values
Self-evident purpose	Appropriateness of name	Measures how appropriate the name is for describing the purpose of the process or practice (consider, for example, whether names are translations or are in a foreign language)	Deceiving, Ambiguous, Partial, Appropriate, Accurate
Self-evident purpose	Recognized purpose	Measures whether new adopters usually recognize the purpose of the process or practice	Yes/No
Learnability	Time required to learn to perform	Measures the time required to learn to perform process or practice activities on tasks of average complexity independently, at a novice level of ability	Number of hours
Learnability	Standard introductory course duration	Measures standard course duration in hours, as defined by authoritative sources	Number of hours
Understandability	# Of specific conceptual definitions	Measures how many specific (new) definitions make up the conceptual model of the process or practice (evaluators must specify the concepts considered)	Number of specific conceptual definitions
Understandability	Conceptual model correspondence	Measures the correspondence between the conceptual model of the process or practice and the user's own conceptual model for the same activity	Low, Medium, High
Understandability	Conceptual model complexity index	Measures the subjective complexity of the conceptual model of the process or practice	Low, Medium, High
Safety	Cost of incorrect adoption	Measures the cost of adopting the process or practice incorrectly as overall impact. Errors include applying the process or practice inappropriately; failing to understand its purpose or dynamics, failure to perform its activities and to evaluate results correctly. For example, incorrect adoption might produce burnout, a high cost, or local inefficiencies, a medium cost	Low, Medium, High

(*continued*)

Table 2. (*continued*)

Characteristic	Metric	Definition	Values
Safety	Reduction in cost of error	Measures how applying the process or practice correctly reduces the overall cost of errors made in the work system. For example, iterative processes are designed to reduce the cost of errors by checking early on, through intermediate results	Low, Medium, High
Safety	Safety perception	Measures how the users perceive the process or practice in terms of safety for themselves and others. For example, if the by-products of executing the process or practice can be used against them, the safety perception might be low	Low, Medium, High
Safety	Use of restraining functions	Measures whether the process or practice provides hard restrictions to prevent the materialization of significant risks	Yes/No
Feedback	Timeliness of feedback	Measures the timeliness of the feedback as perceived by the actor with respect to the action performed and the consequent actions that need to be performed	Immediate, Prompt, Delayed, Non existent
Feedback	Feedback richness	Measures the value of the information received in terms of significance, breadth, depth or nuance	Low, Medium, High
Feedback	People feedback	Measures if the process or practice promotes feedback from people interactions	Yes/No
Feedback	Automatic feedback	Measures if the process or practice provides automatic feedback	Yes/No
Visibility	Defines indicators	Measures if the process or practice defines standard indicators	Yes/No
Visibility	Information tailored to audience	Measures whether information is tailored to better suit different audiences	Yes/No
Controllability	Defines checkpoints	Measures whether the process or practice defines specific checkpoints where users can make decisions that control the outcomes of the process or practice. For example, Scrum Reviews are specific points to evaluate the product and eventually decide whether to accept, reject, or refine a product increment	Yes/No

(*continued*)

Table 2. (*continued*)

Characteristic	Metric	Definition	Values
Controllability	Explicit outcomes	Measures if the process or practice defines outcomes explicitly	Yes/No
Controllability	Level of autonomy	Measures the level of autonomy users have in making decisions related to the execution of the process or practice. Examples include handling unexpected results or deciding whether to proceed or not at specific checkpoints	Low, Medium, High
Adaptability	Defines adaptation points	Measures whether the process or practice defines adaptation points or not. Adaptation points are specific opportunities for variation described by the process or practice. For example, in Scrum the Retrospective is focused on process adaptation	Yes/No
Adaptability	Ratio of roles allowed to adapt	Measures how many roles, from among the process or practice users, are allowed to modify the process or practice, out of the total number of roles (evaluators must specify the roles considered; if no roles are distinguishable, value should be 1)	0 to 1
Attractiveness	User attractiveness rating	Measures how attractive the process or practice is to prospective users (i.e. those lacking experience)	1 to 5
User satisfaction	User experience rating	Measures the subjective experience of using the process or practice	1 to 5

To define a usability profile for a specific process or practice the UMP Evaluation process shown in Table 3 is applied by performing the four activities described.

Table 3. UMP evaluation process.

Activity	Description
Evaluation design	Define the objectives, characteristic and metric exclusions, reference sources and evaluators
Evaluator training	Introduce the usability model and evaluation process to evaluators
Evaluation execution	Perform the evaluation process by analyzing the process or practices according to each sub-characteristic. Determine values for all included metrics according to the analysis performed
Evaluation process review	Complete the evaluation process questionnaire. Review the evaluation results

4 The VMP Study

The study was conducted as preliminary evaluation of UMP utility. The study selected is a naturalistic evaluation according to the classification by Johannesson et al. [13]. A naturalistic evaluation "assesses the artefact in the real world" [13]. The research method selected was the Interview, which "are effective instruments for gathering stakeholder opinions and perceptions about the use and value of an artifact" [13].

The study goal was to evaluate UMP utility in a real-life scenario, specifically whether the evaluation results were valuable to the user.

The VMP method is a participatory approach to milestone planning [20], created to improve the experience of development teams and students who are planning software development projects.

In the Define Requirements activity of the Research Methodology (see Fig. 1) ten potential real-life model application scenarios were defined to help determine applicability (see Appendix). The VMP study was selected because it was an example of one of them, specifically Scenario #8, "Researcher evaluates method, process or practice". This scenario corresponds to an academic context in which a researcher wishes to assess the usability of a process, practice or method. In this case the researcher is the creator of the VMP method. The opportunity for conducting the study arose when the researcher asked the first author to perform an external usability evaluation on the VMP. In this specific situation, given that the researcher required an external evaluation to further his own research activities, the UMP usage by the researcher was restricted to the Profile mode, that is, the researcher used only the evaluation results, and he did not perform the evaluation himself, which was performed by the first author.

The rest of this section is organized as follows: Sect. 4.1 provides an overview of the VMP, while Sect. 4.2 provides details on how the study was conducted.

4.1 VMP Overview

The VMP method is built on top of two existing planning processes, namely Milestone Planning and Participatory Planning [20]. The main VMP contributions are: "The integration of the milestone planning and participatory planning approaches through a visual planning process. A novel construct called the milestone planning matrix, that systematically and visually captures: (1) temporal dependencies between milestones and (2) the allocation of work elements to the milestones they help realize. The reification of work packages by means of sticky notes which must be physically accommodated on a resource and time-scaled milestone scheduling canvas to derive the milestones due dates" [20].

As revealed in [20], student teams in the Master of Software Engineering Program at Carnegie Mellon University have successfully used the VMP for planning their capstone projects, and it has also been taught in several industrial and governmental organizations.

4.2 VMP Study Description

In order to reach the goals of the study, three research questions were posed:

- *RQ1: Is the UMP applicable to the evaluation of the VMP method?*
- *RQ2: Are the UMP model evaluation results helpful in assessing the usability of the VMP method?*
- *RQ3: Is the feedback produced from the UMP evaluation valuable and applicable from the point of view of the VMP creator?*

RQ1 was answered by the feedback from the execution of the UMP Evaluation process by the first author of this paper. An affirmative answer to RQ1 would arise from an effective execution of the UMP evaluation process. RQ2 and RQ3 were answered via a short questionnaire used during the final interview with the VMP creator. Affirmative answers to the questions in the questionnaire would confirm RQ2 and RQ3, as described below.

The VMP study had two roles, researcher (the VMP creator, actor using the evaluation results as described in scenario #8 in Appendix) and evaluator (the first author of this paper who applied the UMP for evaluating the VMP).

The study activities were: (1) Initial definition of expectations of both parties; (2) evaluator (first author) performed VMP evaluation taking as input VMP documentation [20] and information provided by the VMP creator; (3) Evaluator provided feedback to the researcher (an early version of Table 4) who in turn provided minor comments; (4) Final interview where researcher responded to questionnaire; (5) Data analysis and reporting.

The initial interactions were aimed at setting expectations on both parties. Specifically, it was validated with the researcher that the evaluation feedback (VMP usability profile) would take the form of a table with metric values and comments, and that the documentation and interview time from the researcher would be available.

After the initial interactions, the evaluator studied the VMP documentation [20], planned and executed the UMP Evaluation Process on the VMP. Given that the evaluator was the first author of the UMP, the evaluator training activity was not necessary. In the evaluation design all characteristics and metrics were included, although during evaluation some metric values were deemed non-applicable. The execution of the evaluation produced a usability profile with evaluation metrics and comments, presented as feedback to the researcher (as recommended by [28]), who in turn provided confirmation and minor comments. These results are shown in Table 4.

Table 4. VMP usability profile

Metric	Comments	Value
Appropriateness of name	The name describes the essential aspects of the method, that it is visual (and reified), that it is milestone-based and that its purpose is planning	Appropriate
Recognized purpose	From the experiences described by the VMP creator	Yes
Time required to learn to perform	From the experiences described by the VMP creator	4 h
Standard introductory course duration	Informed by the VMP creator	8 h
# Of specific conceptual definitions	Outcomes, Dependencies, Milestone Planning Matrix, Milestone Sequence Diagram, Milestone Effort, Cross-cutting Effort, Milestone Dates, Soft Milestone, Hard Milestone. Milestone work package, Effort unit of time, Milestone scheduling canvas, Milestone list	13
Conceptual model correspondence	It is a participatory planning activity, where the team is responsible for carrying out the plan. The meaning of milestones and due dates is fairly straightforward, as is the rest of the conceptual model	High
Conceptual model complexity index	In general, the data model has low complexity, but specific elements like the pair-wise dependency matrix "roof", the existence of two types of milestones and two types of effort make the overall data model less simple	Medium
Cost of incorrect adoption	It seems hard to use the method so badly that it would produce serious damage	Low
Reduction in cost of error	The focus on milestone planning makes plans "much more stable and practical" than task or activity-oriented plans [20]. The cost of modifying milestones is lower than that of modifying tasks. Making the plan and its elements visual also makes it easier to detect issues and gauge the impact of modifications	High
Safety perception	The team participates in planning its own work. That provides a safer environment for establishing commitments, since these are not imposed from the outside. Depending on the culture of the organization around the team, and the level of autonomy that the team has in planning and executing the plan, the cost of error may vary	High
Use of restraining functions	Matching the scheduling canvas scale to the sticky notes size offers visible hard restrictions on milestone planning to avoid resource over-allocation and help validate milestone viability	Yes

(*continued*)

Table 4. (*continued*)

Metric	Comments	Value
Timeliness of feedback	Creating the Milestones Planning Matrix and the Scheduling Canvas provides early feedback on the soundness of the plan	Prompt
Feedback richness	The feedback confirms that the plan is sound, but does not provide more details	Medium
People feedback	The method does not describe a specific stage to request feedback from others	No
Automatic feedback	Not applicable	No
Defines indicators	The Scheduling Canvas	Yes
Information tailored to audience	Not necessary, the information seems fairly general and without much detail	No
Defines checkpoints	The method describes explicitly several checkpoints during planning	Yes
Explicit outcomes	The Milestone Planning Matrix and the Scheduling Canvas	Yes
Level of autonomy	Teams have a say and are involved, but are not necessarily self-organized	Medium
Defines adaptation points	Milestone sequence diagram is optional	Yes
Ratio of roles allowed to adapt	No roles are defined	Non-applicable
User attractiveness rating	Evaluator opinion after reading the documentation	4
User experience rating	The VMP creator reports anecdotal positive initial responses encountered in both classroom and industry settings. A more precise measurement of satisfaction might provide interesting insights	Not available

The effective evaluation confirmed applicability of the UMP (RQ1) and produced feedback that was presented to the VMP creator.

The questionnaire used during the final interview is shown below, with the corresponding answers:

- Q1: Was the feedback from the evaluation clear and understandable? Yes.
- Q2: Is it useful and applicable in practice? Yes. It was also valuable that the UMP model was already published, and that the UMP first author could act as an external evaluator.
- Q3: Is it coherent with the adoption potential perceived in interactions with method users? Yes.
- Q4: Are you satisfied with the results? Yes.
- Q5: Why? The evaluation touched upon all the main features of the method, and highlighted its contributions.

The analysis of data was very straightforward, given that there was a single data point and the information was aimed directly at evaluating the UMP model. No coding techniques were considered necessary.

The questionnaire answers confirmed that the feedback in the form of the evaluation results was useful and applicable (Q2 for RQ2 and the rest for RQ3). This, together with the manifest initial interest of the VMP creator to have the UMP evaluation performed, provides preliminary confirmation that the UMP was useful in Profile mode; that is, the VMP creator deemed the evaluation results valuable. The VMP creator also valued that the UMP was already published, allowing the UMP to be referenced. It must also be noted that the VMP creator highlighted that the UMP evaluation results touched upon all of the main features of the VMP, hinting that the UMP sensitivity to the VMP was appropriate. In terms of the evaluation results, it is interesting that several salient aspects of the VMP design, such as the reification of work packages as post-it notes and the use of the scheduling canvas as a time-scaled restrictive function, match classical usability principles like affordance and forcing functions and are thus positively highlighted in the evaluation.

The main recommendations provided to the VMP creator were to consider a simplified version of the model for simpler projects and to include some form of satisfaction evaluation in VMP trainings, in order to obtain more systematic feedback.

5 Threats to Validity

In this section the threats to validity of empirical studies are presented, following the categorization provided in [29]:

- Threats to construct validity: for the final interview, this validity may have been affected by the questionnaire design. Care was taken to make answering easy for the respondent, and two authors reviewed and refined the questionnaire.
- Threats to internal validity: In the study, only the VMP creator was interviewed; information about the actual experience of VMP method users is thus not directly available. Future work might include direct measures of the user experience of the VMP, as recommended to the VMP creator. Both the VMP creator and the authors had interests at stake in the study, but the study was carefully designed to reduce bias. For the VMP researcher, the interest at stake was having an external review of the VMP (and possibly a positive evaluation), thus, it did not introduce bias but rather suggests that the UMP evaluation results were applicable. Regarding RQ1 in the study, about UMP applicability to the VMP, the bias of the first author is consistent with the stated interests and typical of Design Science research; evaluation of the UMP by third-parties has been studied in [18] and will be further studied during reliability evaluation.
- Threats to external validity: the bias introduced by limited access to study subjects can have a significant impact on the research. To limit the bias towards accepting any available subjects, the application scenarios for the UMP were defined beforehand. In addition, the ability to generalize from a single preliminary study is very limited, so future studies should encompass other scenarios to improve

generalizability. In particular, the study is an example of the Profile mode, in which only the evaluation results are used; other studies that might assess the Evaluation and Framework modes are needed to further evaluate UMP utility.

- Threats to conclusion validity: the number of observations limits the conclusion validity in this study; further studies for other application scenarios will strengthen the significance of the results. That is why we present this as a preliminary evaluation study, and will expand on it in future work.

6 Conclusions and Future Work

This paper presents a refined version of the Usability Model for Software development Process and Practice (UMP) and a preliminary study for evaluating model utility i.e., its ability to produce valuable results that are useful in a real-life scenario.

The preliminary study results show that the UMP assessment of the usability of the Visual Milestone Planning (VMP) was valued by the VMP creator as an assessment of the VMP contributions and a source of opportunities for improvement. This study was focused on the Profile mode, in which the UMP evaluation performed by the first author provided a usability profile that was used by the VMP creator. In this mode, using the UMP is simpler since evaluations performed by a third-party can be reused, thus reducing the need to perform evaluations, which can be time consuming and require more experience. The VMP creator's interest in having the evaluation also strengthens this preliminary confirmation.

Future research activities include further utility evaluation through other studies that include different scenarios and modes of use (see Appendix) and performing reliability evaluation. For reliability evaluation practitioners will be asked to fill in a survey on applying the model to specific processes and practices, and the inter-evaluator agreement will be assessed on the evaluations, to gauge metric consistency.

Acknowledgements. The research work presented in this paper has been developed within the following projects: the GEMA project ("Consejería de Educación, Cultura y Deporte de la Dirección General de Universidades, Investigación e Innovación de la JCCM", SBPLY/17/180501/000293), the ECLIPSE project ("Ministerio de Ciencia, Innovación y Universidades, y FEDER", RTI2018-094283-B-C31) and the Software Development Process Research Project at the Universidad Nacional de Tres de Febrero (Project lines: Usability of Process and Practice, Agile Practices and Techniques and Requirements Engineering Processes).

Appendix

Supplementary data available at https://doi.org/10.6084/m9.figshare.8292314.

References

1. Conboy, K., Carroll, N.: Implementing large-scale Agile frameworks: challenges and recommendations. IEEE Softw. **36**(2), 44–50 (2019)
2. Kchwaber, K., Sutherland, J.: Scrum Guide. http://www.scrumguides.org/scrum-guide.html. Accessed 08 June 2019
3. Paez, N., Fontdevila, D., Gainey, F., Oliveros, A.: Technical and organizational Agile practices: A Latin-American survey. In: Garbajosa, J., Wang, X., Aguiar, A. (eds.) XP 2018. LNBIP, vol. 314, pp. 146–159. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91602-6_10
4. Kuhrman, M, et al.: Hybrid software development approaches in practice: a European perspective. IEEE Softw. (2018)
5. Ambler, S.: Agile practices survey results, July 2009. http://www.ambysoft.com/surveys/practices2009.html. Accessed 08 June 2019
6. Chow, T., Cao, D.B.: A survey study of critical success factors in agile software projects. J. Syst. Softw. **81**(6), 961–971 (2008)
7. Forsgren, N., Humble, J., Kim, G.: Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations. IT Revolution Press, Portland (2018)
8. Graziotin, D., Wang, X., Abrahamsson, P.: Software developers, moods, emotions, and performance. IEEE Softw. **31**(4), 24–27 (2014)
9. Riemenschneider, C.K., Hardgrave, B.C., Davis, F.D.: Explaining software developer acceptance of methodologies: a comparison of five theoretical models. IEEE Trans. Softw. Eng. **28**(12), 1135–1145 (2002)
10. Overhage, S., Schlauderer, S., Birkmeier, D., Miller, J.: What makes IT personnel adopt scrum? A framework of drivers and inhibitors to developer acceptance. In: The Proceedings of the Annual Hawaii International Conference on System Sciences (2011)
11. Kroeger, T.A., Davidson, N.J., Cook, S.C.: Understanding the characteristics of quality for software engineering processes: a Grounded Theory investigation. Inf. Softw. Technol. **56**, 252–271 (2014)
12. Feiler, P., Humphrey, W.: Software process development and enactment: concepts and definitions. Software Engineering Institute, CMU/SEI-92-TR-004 (1992)
13. Johannesson, P., Perjons, E.: An Introduction to Design Science. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-319-10632-8
14. International Organization for Standardization, ISO/IEC 25010 Systems and Software Engineering - Systems and Software Quality Requirements and Evaluation (SQuaRE) - System and Software Quality Models, Geneva, Switzerland (2011)
15. Norman, D.A.: The Design of Everyday Things. Basic Books, New York (1988)
16. Nielsen, J.: Usability Engineering. Elsevier, Amsterdam (1994)
17. International Organization for Standardization: ISO/IEC 25040 Systems and Software Engineering – System and software Quality Requirements and Evaluation (SQuaRE) – Evaluation process, Geneva, Switzerland (2011)
18. Fontdevila, D., Genero, M., Oliveros, A.: Towards a usability model for software development process and practice. In: Felderer, M., Méndez Fernández, D., Turhan, B., Kalinowski, M., Sarro, F., Winkler, D. (eds.) PROFES 2017. LNCS, vol. 10611, pp. 137–145. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69926-4_11
19. Kontio, J., Bragge, J., Lehtola, L.: The focus group method as an empirical tool in software engineering. In: Shull, F., Singer, J., Sjøberg, D.I.K. (eds.) Guide to Advanced Empirical Software Engineering, pp. 93–116. Springer, London (2008). https://doi.org/10.1007/978-1-84800-044-5_4

20. Miranda, E.: Milestone Planning: A Participatory and Visual Approach (2018). https://www.researchgate.net/publication/328918275_A_Participative_Visual_Approach_to_Milestone_Planning, https://doi.org/10.13140/rg.2.2.18969.06241. Accepted to be published in The Journal of Modern Project Management
21. Culver-Lozo, K.: The software process from the developer's perspective: a case study on improving process usability. In: Proceedings of the Ninth International Software Process Workshop, Airlie, VA, pp. 67–69 (1994)
22. Cockburn, A.: Agile Software Development: The Cooperative Game. Pearson Education, London (2006)
23. Franch, X., Carvallo, J.P.: Using quality models in software package selection. IEEE Softw. **20**(1), 34–41 (2003)
24. Dreyfus, S.E., Dreyfus, H.L.: A five-stage model of the mental activities involved in directed skill acquisition. University of California/Berkeley Operations Research Center. DTIC ADA084551 (1980)
25. Basili, V, Caldiera, G., Rombach, H.D.: The goal question metric approach. In: Encyclopedia of Software Engineering. Wiley (1994)
26. Fenton, N.E., Pfleeger, S.L.: Software Metrics: A Rigorous and Practical Approach. CRC Press, Boca Raton (1997)
27. International Organization for Standardization: ISO/IEC 15939 Systems and Software Engineering – Measurement process, Geneva, Switzerland (2017)
28. Runeson, P., Höst, M.: Guidelines for conducting and reporting case study research in software engineering. Empirical Softw. Eng. **14**, 131 (2009)
29. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in Software Engineering. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29044-2