

Johansen, J., Colomo-Palacios, R. and O'Connor, R.V., 2016, June. Towards a Manifesto for Software Process Education, Training and Professionalism. In International Conference on Software Process Improvement and Capability Determination (pp. 98-105). Springer International Publishing.

Towards a Manifesto for Software Process Education, Training and Professionalism

Jørn Johansen ¹, Ricardo Colomo-Palacios², Rory V. O'Connor ³

¹ Whitebox, Horsholm, Denmark
jj@whitebox.dk

² Ostfold University College, Norway
Ricardo.colomo-palacios@hiof.no

³ Dublin City University, Dublin, Ireland
Rory.OConnor@dcu.ie

Abstract. In June 2015 a group of experts in Software Process Improvement (SPI) and Education from all over the world gathered at the 1st International Workshop on Software Process Education, Training and Professionalism held in connection with 15th International Conference Software Process Improvement and Capability Determination. Discussions with key players in the relevant professional and personal certification fields, as well as experienced educators led to a consensus that it is time for the industry to rise to the new challenges and set out in a manifesto a common vision for educators and trainers together with a set of recommendations to address the challenges faced. At the workshop 14 “experts” from education and industry presented and discussed their “wisdom and experience” of the challenges faced for software process education, training and professionalism, especially with the background of the new modes of learning and teaching in higher education. Based on the presentations, 32 workshop participants brainstormed core values and principles specifically addressing the needs of software process education, training and professionalism. Via affinity analysis and group thinking exercises we identified an initial manifesto, consisting of 10 values and 4 principles. It is expected that this draft manifest will give expression to state-of-the-art knowledge on software process education, training and professionalism. It is based on hundreds of person-years of practice and experience from educators and industry professionals globally. Further work is currently being undertaken to extend and validate this draft manifesto with a view to publishing in its entirety by 2016.

Keywords: Software Process; Education; Training; Professionalism.

1 Introduction

Within the broad field of software engineering, and according to SWEBOK [1], software engineering processes or software processes in short are concerned with work activities accomplished by software engineers to develop, maintain, and operate software, such as requirements, design, construction, testing, configuration management, and other software engineering processes. Software Process is one of

the fifteen knowledge areas (KA) defined in SWEBOK 3.0 [1] and was also one of the ten KAs defined in the previous version of this body of knowledge.

The software process is concerned with software process definition, software life cycles, software process assessment and improvement, software measurement, and software engineering process tools. Software process is inherent to software practice. In working scenarios, software practitioners are often unfulfilled with their level of preparation when they start their careers [2]. Literature pointed out that, among other aspects, this problem lies in the way software process is typically taught at universities [3]. These courses present constraints inherent in an academic setting including depth and time limitations. These restrictions lead to inefficient training in the many facets of the software lifecycle [4].

However and in spite of these recent and disappointing studies, the topic is covered in current curricular efforts in the fields of software engineering. In the field of undergraduate degree programs, according to the Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering [5], software process is one of the 10 knowledge areas of the curriculum. In this publication, authors identify 467 hours of course contents and 10 courses or knowledge areas. Software Process course is presenting a teaching load of 33 hours covering various process models that support individual and team experiences with one or more software development processes, including planning, execution, tracking, and configuration management. Moreover, one of the guidelines in the curriculum definition claims, "Software process should be central to the curriculum organization and to students' understanding of software engineering practice". In the field of graduate degree programs, the equivalent effort is the Curriculum Guidelines for Graduate Degree Programs in Software Engineering launched back in 2009 [6]. In this case, Software Process is one of the 8 knowledge areas in the Software Engineering section of the body of knowledge. In a nutshell, the difference between the undergraduate and graduate curricular effort in the software process arena is rooted on the fact that in the first the topic is addressed only at Bloom's Taxonomy levels 1 and 2 while in the second it is covered at levels 2 and 3.

Although the coverage of software process education is established in curriculum initiatives, increasing its coverage in educational settings is still challenging. The complexity of the subject together with the need of a good background of the discipline is normally pushing subjects into master programs, while personal and team software approaches are mostly present in bachelor curricula [7]. Further as has been noted by Prof. Margaret Ross, the UK's most influential software quality educator and commentator, there is a lack of relevant knowledge and experience of teachers and lecturers coupled with the problems of pressures by other topics on academic course [8]. Further she states that most syllabuses are already very full on these courses, with constant pressure to introduce additional topics. Dedicated units on quality and process improvement are not usual.

An associated aspect of software process education and professionalism is related to the teaching and usages of international standards in educational settings. Whilst there have been limited attempts to teach international software process standards to students [15] and engage professionals [16, 19] alike, these have met with limited success [18].

Quite apart from the issues of education and training, it has been well established that there are business benefits to the adoption of SPI practices in an industrial setting,

although some practice issues remain in some areas such as SME sector [5, 6, 17]. Moreover, recent studies e.g. [9] showed up gaps in software process competence in samples of software professionals. It is therefore considered both appropriate and necessary to expand the remit of an SPI education manifesto beyond the realm of education and training and to include professionalism from an industry practitioner perspective as well.

As a result of this need, a set of software process consultant and practitioners along with a group of academics on the topic detected this gap and decided to launch a manifesto for software process education, training and professionalism. This paper is devoted to illustrate the initial steps taken by these set of experts towards this goal. The remaining of the paper is structured as follows. In Section 2, authors present the initiative and initial results along with ongoing works. Section 3 is aimed to wrap up the paper and to portray main future works.

2 A manifesto for software process education, training and professionalism

According to the Merriam Webster dictionary, a manifesto is a written statement declaring publicly the intentions, motives, or views of its issuer. The etymological origin of the word, according to Oxford dictionaries can be rooted in the Italian (mid-17th century) from *manifestare* that is also from Latin, 'make public', from *manifestus*. Manifestos are quite common in the technological arena. Maybe the most popular of them is the Manifesto for Agile Software Development [10] while there are other with less repercussion like, for instance, the Manifesto for Software Craftmanship [11] or the SPI Manifesto [12].

Following the path previously followed by these initiatives, in this section authors explain the process, structure and initial results for the Manifesto for Software Process Education, Training and Professionalism.

2.1 The process

In June 2015 a group of 32 experts in Software Process Education, Training and Professionalism from 15 different countries and 3 continents gathered in connection with the SPICE 2015 Conference [13] for a workshop at Gothenburg University in Sweden. This workshop was the 1st International Workshop Software Process Education, Training and Professionalism (SPEPT 2015) [14]. The initial aim for the workshop was to present a set of works on the topic but, taking into account the importance of the topic and previous feedback from scientific and professional arenas, it was aimed to develop a manifesto for Software Process Education, Training and Professionalism. The overall structure of the manifesto is based on previous initiatives on the topic and more precisely, the SPI Manifesto, launched by the EuroSPI community by 2009 [12]. In what follows, the main aspects of the process of the definition of the manifesto is presented.

- A. Before the workshop, facilitator performed a set of tasks to support the definition of the manifesto:
 - I. Workshop facilitator extracted 7 preliminary values and 139 principles derived from 10 background papers describing the problems and barriers in SE and SPI education and learning presented in the workshop.
 - II. These principles were group into 24 different topics.
 - III. The 24 topics was allocated to 4 working groups: Method and Delivery; Certification and Training; Links between Management and SE/SPI and, finally, SPI.

As a final remark, authors want to underline that it took much more time than planned to identify the preliminary values and principles from the papers, and bring it to a form, which enabled it for the workshop as basis material.

- B. During the workshop:
 - IV. In the morning session ten live presentations on the topic were scheduled and a set of recorded videos supporting the initiative were displayed.
 - V. In the evening session, a short description of the workshop tasks was presented to workshop attendants by workshop facilitator.
 - VI. The set of materials to develop the task were presented. Materials include stickers, wall papers, pens, postIts, labels with the 7 values, labels with the allocated principles.
 - VII. Participants were divided into four different groups to develop values and practices along as identifying supporting actions according to the 4 working groups defined earlier. Each group presented a moderator.
 - VIII. The groups first discussed the initial values. It was allowed to come up with new values.
 - IX. Than the groups sorted and grouped the allocated principles. It was allowed to remove and to come up with new principles.
 - X. The grouped principles was dot voted and the most important were linked to the values.
 - XI. Apart from the set of values and principles, all groups also proposed a set of supporting actions for the initiative.
 - XII. Once defined, a short presentation on the outcomes of each group was provided, which was recorded on video.
 - XIII. Facilitator documented the process and presented an overall preliminary result of the workshop at the end of the conference.
- C. After the workshop.
 - XIV. The editorial board for the manifesto is designed and contributors join editors in specific areas.
 - XV. A template of the manifesto, the background papers, the initial values and principles as well as the documentation from the workshop were sent to the editorial board.
 - XVI. Editors and contributors develop editorial content.
 - XVII. A number of iterations of writing, reviewing, commenting and rewriting took place.

- XVIII. Expert reviews are provided and the document was updated.
XIX. Final document is edited and distributed.
The whole process is scheduled to be complete by 2016.

2.2 The structure

As stated before, the structure of the manifesto is based on the SPI Manifesto [12]. Consequently, the manifesto adopts the approach based on values and principles.

Value or Values present several entries in dictionaries, but focusing in the intended meaning for this document, values are principles or standards of behavior; one's judgement of what is important in life, according to the Oxford Dictionary. In the Cambridge dictionary, the term is defined as the beliefs people have, especially about what is right and wrong and what is most important in life, that control their behavior. For the aims of the manifesto values represent the core priorities in an education culture, including what drives priorities and how you truly act when doing education. In other words, a value is something that deserves to be in focus because of its importance or worth. The identified values is core for Software Engineering and Software Process Improvement are the values that we have prioritized.

Principle presents also different meanings. According to Merriam-Webster dictionary, a principle is a moral rule or belief that helps you know what is right and wrong and that influences your actions. Oxford dictionaries offer a definition of the term as follows: a fundamental truth or proposition that serves as the foundation for a system of belief or behavior or for a chain of reasoning. Again, for the sake of this manifesto, a principle is a basic generalization that is accepted as true and that can be used as a basis for education reasoning or education behavior. A principle is something that can serve as a foundation for action to reach the value. You can use the principles to govern your personal behavior in relation to reach the necessary competences for Software Engineering and Software Process Improvement work.

Finally, practices are specific supporting actions for principles and values.

2.3 Initial results

It is important to note that, although process was defined and explained to participants, the documentation of the work differed to some degree among groups. This difference introduce some extra work for the editors while writing the parts of the manifesto – the complete overview was difficult to keep when going into detail of the documentation of the workshop.

Results from the workshop with regards to groups, 10 values, 15 principles and supporting actions are as follows:

VALUES

1. Professional achievement: Experience a feeling of accomplishment for job well done, that you have made a contribution, sense of competence.

2. Knowledge management: Feel what you do makes a difference, provides new knowledge, enhances existing systems, provides development of others.
3. Personal competence: Engage in work that offers opportunity to learn and grow as a person, room for retrospective.
4. Universal recognition: Have others look up to you, admire your skill and expertise, be seen as admirable and successful, have a sense of knowledge and prestige.
5. SPI as profession: The conduct, aims, or qualities that characterize or mark a profession or a professional person"; and it defines a profession as "a calling requiring specialized knowledge and often long and intensive academic preparation.
6. Innovative: Focused on constant improvement and being at the forefront of change and innovation in education and training.
7. Accessibility: Providing a framework of flexible learning opportunities, proactive support mechanisms and administrative processes facilitating and simplifying access for students at all levels and facilitating the transferability of credits.
8. Value: Value to the business according to customers.
9. Inspiring: No consensus was reached on its definition. Final definition will be provided in future steps.
10. Collaboration: No consensus was reached on its definition. Final definition will be provided in future steps.

PRINCIPLES:

During the workshop, 3 to 4 principles per group were defined. These principles are in writing at the moment, but the link between group and principles is as follows:

1. Link between management & SE/SPI: Evaluation; Content; Form.
2. Method & delivery: Technical approaches; Industry collaboration; tools; Learning approaches.
3. Certification & training: Value added; Holistic; Innovation; Just good enough.
4. SPI: Content; Lean by doing; Form; Model.

SUPPORTING ACTIONS:

- Get professional societies unified
- Create Body of Knowledge
- Develop a SPI Book of Knowledge
- Link to professional association (IEEE, ICM, ISTQB)
- Consult with Certification Scenes (e.g. ARCS, ECQA, ISTQB)
- Link to HR associations / Skills (ECTS)
- Investigate systems that measure experience levels

3 Conclusions

The development of a Manifesto for Software Process Education, Training and Professionalism has the ability to assist with addressing many of the identified issues and gaps facing both educators and the software profession today

As there is always pressure to include new topics on courses, the professional bodies, they could specify that process improvement should be included in any degree course to be accredited by that body. The governments, through their financial power, could play a major role in encouraging the professional bodies and the universities and colleges to give a higher priority to relevant courses and in particular to quality and process improvement

To assist the lecturers to inspire their students, in addition to helping with suitable material, opportunities could be provided for lecturers and teachers to gain real life experience by shadowing process improvement professionals, possibly with Certification Bodies, subject to their clients' agreement, and in organisations with quality and process improvement sections. This would enable the lecturers to introduce some real world, even though limited, experience to their discussions with students.

From a professional perspective, there are limited numbers of professionals with adequate experience and knowledge of process improvement, to be able to influence the majority of organisations. In many cases, there is little opportunity of gaining practical experience, especially if they are employed by SMEs. Individual (professionals) could address these problems by attending professional training and University courses.

Software process improvement is considered one of the most important fields in the software engineering discipline. However, and in spite of its importance, increasing its coverage in educational settings is still challenging. By influencing the syllabus for these courses, and other degree courses, to include quality and process improvement, the future professionals, on entering the various Industries, could act as ambassadors for process improvement for the future. Ultimately, it is the hope of the champions behind the development of the Manifesto for Software Process Education, Training and Professionalism that it can address some of these needs.

Acknowledgments: Authors would like to thank all workshop participants and manifesto reviewers and editors for their support to the initiative.

References

1. Abran, A., Fairley, D.: SWEBOK: Guide to the software engineering Body of Knowledge Version 3. IEEE Computer Society (2014).
2. Exter, M.: Comparing Educational Experiences and On-the-job Needs of Educational Software Designers. Proceedings of the 45th ACM Technical Symposium on Computer Science Education. pp. 355–360. ACM, New York, NY, USA (2014).
3. Ali, N. bin, Unterkalmsteiner, M.: Use and evaluation of simulation for software process education: a case study. Proceedings of the European Conference Software Engineering Education. pp. 59–73. Shaker Verlag, Seon Monastery, Germany (2014).

4. Kohwalter, T.C., Clua, E.W.G., Murta, L.G.P.: Reinforcing Software Engineering Learning through Provenance. 2014 Brazilian Symposium on Software Engineering (SBES), pp. 131–140 (2014).
5. LeBlanc, R.J., Sobel, A.: Software Engineering 2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. IEEE Computer Society (2014).
6. Adcock, R., Alef, E., Amato, B., Ardis, M., Bernstein, L., Boehm, B., Bourque, P., Brackett, J., Cantor, M., Cassel, L., others: Curriculum Guidelines for Graduate Degree Programs in Software Engineering. Stevens Institute of Technology (2009).
7. Heredia, A., Colomo-Palacios, R., Amescua-Seco, A.: A Systematic Mapping Study on Software Process Education. Proceedings of 1st International Workshop on Software Process Education, Training and Professionalism. pp. 7–17. Ceur Workshop Proceedings, Gothenburg, Sweden (2015).
8. Ross, M.: Process Improvement - Barriers and Opportunities for Teaching and Training. Proceedings of 1st International Workshop Software Process Education, Training and Professionalism (SPEPT 2015), EUR Workshop Proceedings Series, Vol. 1368. EUR Workshop Proceedings Series (2015).
9. Colomo-Palacios, R., Casado-Lumbreras, C., Soto-Acosta, P., García-Peñalvo, F.J., Tovar-Caro, E.: Competence gaps in software personnel: A multi-organizational study. *Computers in Human Behavior*. 29, 456–461 (2013).
10. Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., others: Manifesto for agile software development. (2001).
11. Ambler, S.: The Manifesto for Software Craftsmanship, <http://manifesto.softwarecraftsmanship.org/>.
12. Pries-Heje, J., Johansen, J.: SPI Manifesto, http://www.madebydelta.com/imported/images/DELTA_Web/documents/Ax/SPI_Manifesto_A.1.2.2010.pdf, (2010).
13. Rout, T., O'Connor, R.V., Dorling, A.: Software Process Improvement and Capability Determination. *Communications in Computer and Information Science* Vol. 526, Springer (2015).
14. O'Connor, R.V., Mitasiunas, T., Ross, M.: Proceedings of 1st International Workshop Software Process Education, Training and Professionalism. EUR Workshop Proceedings Series, 1368 (2015).
15. Laporte, C.Y. and O'Connor, R.V., Software Process Improvement in Graduate Software Engineering Programs, proceedings 1st International Workshop Software Process Education, Training and Professionalism (SPEPT 2015), pp. 18 - 24, CEUR Workshop Proceedings Vol. 1368, Gothenburg, Sweden, 2015.
16. Laporte, C. Y., O'Connor, R. V., Garcia Paucar, L. and Gerancon B., An Innovative Approach in Developing Standard Professionals by Involving Software Engineering Students in Implementing and Improving International Standards, *Journal of the Society for Standards Professionals*, Vol. 67, No. 2, 2015.
17. Larrucea, X., O'Connor R.V., Colomo-Palacios, R., and Laporte, C.Y., Software Process Improvement in Very Small Organizations, *IEEE Software*, Vol. 33, No. 2, pp. 85-89, March/April 2016.
18. O'Connor, Rory V., Developing software and systems engineering standards, In *Proceedings of the 16th International Conference on Computer Systems and Technologies*, pp. 13-21. ACM, 2015.
19. O'Connor, R.V., Laporte, C.Y.: An Innovative Approach to the Development of an International Software Process Lifecycle Standard for Very Small Entities. *International Journal of Information Technology Systems Approach*, 7, 1–22 (2014)