Muñoz M., Peña A., Mejia J., Gasca-Hurtado G.P., Gómez-Alvarez M.C., Laporte C. (2019) A Comparative Analysis of the Implementation of the Software Basic Profile of ISO/IEC 29110 in Thirteen Teams That Used Predictive Versus Adaptive Life Cycles. In: Walker A., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2019. Communications in Computer and Information Science, vol 1060. Springer, Cham

A comparative analysis of the implementation of the Software Basic profile of ISO/IEC 29110 in thirteen teams that used predictive versus adaptive life cycles

Mirna Muñoz [1], Adriana Peña[2], Jezreel Mejia[1], Gloria Piedad Gasca-Hurtado [3], María Clara Gómez-Alvarez[3] and Claude Y. Laporte[4]

[1] Centro de Investigación en Matemáticas, Parque Quauntum, Ciudad del Conocimiento
Avenida Lassec, Andador Galileo Galilei, Manzana, 3 Lote 7 CP 98160. Zacatecas, Zac.
{mirna.munoz,jmejia}@cimat.mx

[2] Departamento de Ciencias de la Computación, CUCEI de la Universidad de Guadalajara,
Av. Revolución No. 1500, 44430, Guadalajara, Jal., México
adriana.pena@cucei.udg.mx

[3] Universidad de Medellín, Maestría en Ingeniería de Software, Facultad de Ingenierías, Carrera 87 No. 30-65, Medellín, Colombia
{gpgasca, mcgomez}@udem.edu.co

[4] École de technologie supérieure, Department of Software and IT Engineering
1100, Notre-Dame Street West, Montréal, Québec, Canada, H3C 1K3
Claude.Laporte@etsmtl.ca

Abstract. The growing of the software demand has created opportunities for software development organizations. Especially for very small entities (VSEs), because a large number of them produce software for medium and large companies as part of a production chain that provides products and services to satisfy market needs. This situation highlights the increasing need for improving their software development processes in order to stay in business by developing quality software products and services with limited resources. Unfortunately, a common issue faced by most of the VSEs is the lack of knowledge and practical experience regarding the implementation of current SPI models and software engineering standards. In this context, the ISO/IEC 29110 series of standards and guides have been developed to help VSEs to improve their development processes. One of the main features of this standard is that it can be adapted to the software lifecycle of VSEs. This paper presents the results of a comparative analysis of 13 software organizations with teams that work using predictive methodologies and those using adaptive methodologies during the implementation of the software Basic profile of the ISO/IEC 29110. The results show that after executing the method teams have a high level of coverage regarding the ISO/IEC 29110.

Keywords: ISO/IEC 29110, very small entities, software development team, process improvement, adaptive methodologies, predictive methodologies, lifecycle, Basic profile.

1 Introduction

Software standards and models for the software industry are elaborated to contribute to the development of quality products within budget and schedule, by optimizing efforts and resources. However, the implementation of proven practices contained in these models and standards in real environments of software development organizations represents an actual challenge. Especially for very small entities (VSEs), i.e. enterprise, organization (e.g. public or non-profit organization), project or department having up to 25 people, that must work harder in order to survive, and at the same time spending time and effort on improving their operation and processes [1].

The ISO/IEC, particularly the Working Group 24, provides solutions to help VSEs to implement proven practices such as the ISO/IEC 29110 series of standards and guides. These series aim to help VSEs to improve their system or software development process, helping them in the implementation of proven practices in order to get benefits such as increasing their product quality, reducing their development time, the percentage of rework and develop their product within budget and schedule.

Also, a common problem that most VSEs face is the lack of knowledge and practical experience regarding the implementation of software models, such as the CMMI® [2], or software engineering standards such as the ISO/IEC/IEEE 12207 lifecycle processes standard [3].

This paper presents the results of the implementation of the software Basic Profile of the ISO/IEC 29110 in 13 VSEs of Mexico composed of teams from 2 to 6 people and using two types of life cycles (i.e. predictive and adaptive) [4].

This analysis was done, since one of the main features of ISO/IEC 29110 is that it can be implemented in VSEs using any development approach or methodology including, for example, agile, evolutionary, incremental, or test-driven development, among others [5]. This paper aims to identify the effort invested by software teams in order to improve their processes using the software Basic profile of ISO/IEC 29110 as a framework.

After the introduction, this paper is structured as follows: section 2 shows the background of this research composed of key concepts; section 3 shows the method followed by the teams to implement the software Basic profile of ISO/IEC 29110; section 4 presents the comparative analysis of the teams' performance during the implementation of the software Basic Profile of ISO/IEC 29110 from the beginning to their certification; and finally, section 5 presents discussion and conclusions.

2 Background

2.1 Software Development Teams

A team is a group of people performing together a set of activities with a common interest. The team members have a common objective, performing in an autonomous way and coordinating by themselves [6].

Nowadays, the growth on software requirements has made its development to be essentially performed by a team. A team is more than just people working together [7]. Authentic teamwork requires from its team members a professional, but also a personal relationship. The software development team is considered a socio-technical endeavor; it requires problem solving capabilities, cognitive aspects and social interaction. Better results can be achieved when people with particular social skills are assigned to different phases of a project, those that best match their skills [8]. This intrinsic relation among social and technical skills makes it difficult to understand why some teams are less successful than others.

2.2 ISO/IEC 29110 series

The ISO/IEC 29110 series of software engineering standards and guides have been developed to assist VSEs to improve their systems or software development process. In the context of ISO/IEC 29110, systems are typically composed of hardware and software components. The ISO/IEC 29110 series, more specifically the management and engineering guides, should help VSEs in the implementation of the proven practices, in order to get advantages such as increasing their product quality, reducing their development time and help them develop their product within budget and schedule.

Some of the ISO/IEC 29110 main features are: (1) it provides a set of 4 software profiles to be used by the VSEs according to their goals: *Entry profile, Basic profile, Intermediate profile and Advanced profile.* The software Basic profile is the only profile, at the moment, to which a VSE can be certified; (2) it provides two main categories of process, the project management process and the software implementation process; (3) it can be used to establish processes in VSEs using any development approach or methodology and (4) it provides a set of process elements such as objective, activities, tasks, roles and work products.

The software Basic profile is composed of two processes [9, 10]:

- *The Project Management process*: it aims establishing and carrying out the tasks related to a project management in a systematic way, so that the project's objectives are completed with the expected quality, time and costs. It has 4 activities: project planning, project plan execution, project assessment and control and project closure.
- *The Software Implementation process*: it aims performing in a systematic way, of the activities related to the analysis, design, construction, integration and test, according to the requirements speci-

fied of new or modified software products. It has 6 activities: initiation, analysis, design, construction, integration and test and delivery.

2.3 Software Development Life Cycles

According to the Guide to the Project Management Book of Knowledge (PMBOK® Guide) [11], in projects involving more than one phase, there should be an analysis of the features such as the level of control required, effectiveness and degree of uncertainty to determine a correct relationship between individual phases, for example: sequential, parallel, etc.
In this paper we will compare teams that use predictive life cycles to teams that use adaptive life cycles as next described:

- *Predictive life cycles methodologies*: is based on development cycles in which the project scope, time and cost required to be defined in order to achieve the task. The cycles are determined as early in the development cycle as possible. The projects are performed through a series of sequential or overlapping phases, in which each phase is focused on a subset of project activities.
  The main features of teams using these life cycles are: (1) they focus on defining the overall scope for the product and project; (2) they develop a plan to develop a product and any associated deliverables; (3) they proceed to execute the plan through the phases to achieve the goal; (4) they manage carefully the changes and the required re-planning and a formal acceptance. Examples of these life cycles are: Team Software Process (TSP®) and waterfall.
- *Adaptive life cycles methodologies*: is based on the development cycles in which the overall scope of the project will be decomposed into a set of requirements as well as work to be performed to achieve the goal. This life cycle intent to respond to high levels of change while increases the stakeholder involvement. The projects are performed in iterations in which generally several processes are performed.
  The main features of teams using these life cycles are: (1) they focus on decomposing the goal of the project in requirements and work to perform and allocate it in a product backlog; (2) they determine the work to be done in the next iteration, selecting items from the product backlog; (3) they proceed to execute the select items so that, at the end of the iteration, they can be able to present a product ready to be reviewed by the customer; (4) they intent to respond to a high number of changes. An example of this life cycle is Scrum.

3 Method for implementing the ISO/IEC 29110 series

To perform the adoption and implementation of the software Basic profile of ISO/IEC 29110 by the teams of the 13 VSEs, they followed a six-step method [12] depicted in Figure 1. It is important to mention that the steps were adapted, by changing the acronym VSE for teams, in order to highlight that the method was implemented by the teams.

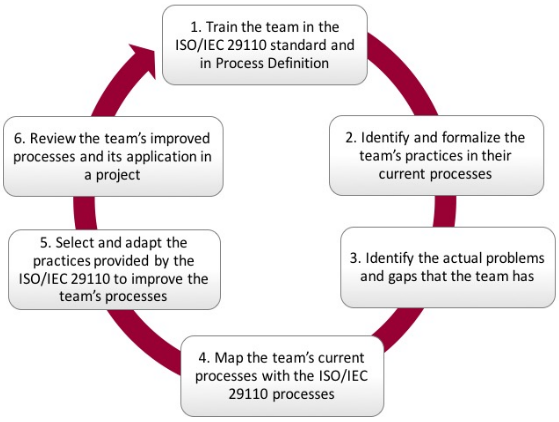
1. Train the team in the ISO/IEC 29110 standard and in Process Definition
2. Identify and formalize the team's practices in their current processes
3. Identify the actual problems and gaps that the team has
4. Map the team's current processes with the ISO/IEC 29110 processes
5. Select and adapt the practices provided by the ISO/IEC 29110 to improve the team's processes
6. Review the team's improved processes and its application in a project

Fig. 1. Six-step method adapted from [12] followed by each team during the adoption of the software Basic profile of ISO/IEC 29110.

- Step 1. Train the team in the Basic profile of ISO/IEC 29110 and in the process definition by performing a set of workshops and seminars as follows: two sessions, totaling 15 hours, focused on the ISO/IEC 29110 in which the concepts, the structure, the processes of the software Basic profile of ISO/IEC 29110 and some examples of its implementation are presented and discussed and two sessions, totaling 15 hours, focused on process definition in which the process elements, identification of best practices of their own processes and the formalization of their own processes. So, they can be aware of the use of the standard as well as the process description.
- Step 2. Identify and formalize team's practices in their current processes, this step is focused on formalize the practices for both processes: project management and software implementation.
- Step 3. Identify problems and gaps that the team has. This step is focused on identifying the problems the team has with their actual way of work, regarding the project management and project implementation process.
- Step 4. Map the team's current processes with the ISO/IEC 29110 processes, to perform the mapping between the team's current process and the Basic profile of ISO/IEC 29110. The mapping is performed for both processes (Project Management process and Software Implementation process).
- Step 5. Select and adopt the practices provided by the Basic profile of ISO/IEC 29110 to improve the team's processes, according to the identified problems and gaps.
- Step 6. Review the team's improved processes as well as their implementation in a project. Besides, the non-conformities with respect to the standard are reported.

It is important to mention that to implement the method, a set of meetings were performed. Also, three milestone meetings were held to assess the practices coverage allowing us to perform the comparative among teams, as follows described: (milestone 1): we call this milestone initial diagnostic. This milestone was set after the execution of four steps of the method, because at that time teams have their current processes defined; (milestone 2): we call this milestone middle diagnostic. This milestone was set after the execution of the step five of the followed method, because at that time teams start the adoption of ISO/IEC 29110 to their processes and according to their way of work; (milestone 3): we call this milestone final diagnostic. This milestone was set after the execution of the sixth step of the method, because at that time teams were ready to start a formal audit process to be certified to the software Basic profile of ISO/IEC 29110.

4 Comparison in the implementation of the Basic profile of ISO/IEC 29110

In this paper, we are making a comparison by isolating other factors and considering only the followed method on a quantitative analysis. The analysis aims to identify differences and similarities among teams to understand the impact of following a method to implement the Basic profile of the ISO/IEC 29110. To carry out the analysis we established the coverage levels as next described: *high level* (from 46-67 number of practices); *medium level* (from 23-45 number of practices); *low level* (from 0-22 number of practices).

4.1 Teams description

Table 1 shows the description of the 13 teams, who were trained in a four months' period, 7 teams in 2017 and 6 teams in 2018. The first column provides the team ID, second column presents the number of members of the team, the third column describes the type of methodology or model used by that team, the fourth column lists the software product selected for the ISO/IEC 29110 certification to the Basic profile. The fifth column lists the number of meetings held with each team (the training sessions and the workgroup meetings are not included) and, the sixth column presents the year in which the method was implemented for each team.
It is important to highlight that E7 and E10 teams had fewer meetings because both teams had previous experience with the use of models and standards such as CMMI® and Moprosoft.

Table 1. Teams description

Team ID	# of members	Methodology	Developed product	# of meetings	Trained period
E1	2	None, but using some agile practices	Embedded software	5	2017
E2	5	Hybrid: TSP-Scrum	Automation of the quali-	6	2017

			ty management system		
E3	5	Scrum	System of inventory control	5	2017
E4	4	Scrum	Medical consultation management	5	2017
E5	5	Scrum	System for a fitness center management	5	2017
E6	5	Scrum	Insurance management	5	2017
E7	5	Hybrid: CMMI®-Scrum	Transport management system	2	2017
E8	5	Waterfall	Registration system for research projects	5	2018
E9	4	TSP	Social service monitoring system	6	2018
E10	3	Waterfall	Chemistry laboratory system	2	2018
E11	4	Hybrid: CMMI®-Scrum	Livestock system	4	2018
E12	5	Methodology based on CMMI® model	Professors assessment system	5	2018
E13	4	TSP	Access control to Linux laboratory	6	2018

4.2 Initial diagnostic comparison

The initial diagnostic comparison was performed after the execution of the steps 1, 2, 3 and 4 of the method used and corresponds to the milestone 1. Figure 2 summarizes the data of the practice coverage.

As Figure 2 shows, most of the teams have a high and medium percentage of practices not covered regarding the Basic profile of ISO/IEC 29110. If we analyze the number of not covered practices: 4 teams (31%) have from 46 to 67 practices not implemented; 8 teams (61%) have from 23 to 45 not implemented practices and 1 team (8%) has from 0 to 22 not implemented practices.

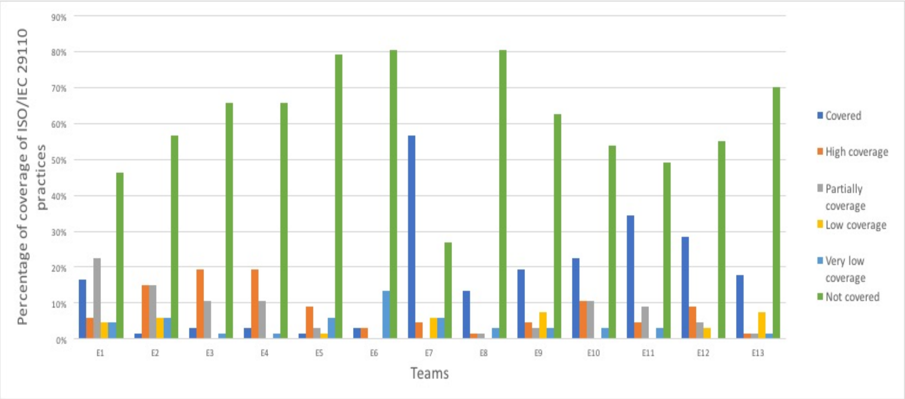

Fig. 2. Milestone 1: initial diagnostic comparing the results after the implementation of the fourth step of the method used.

To better understand the performance of the team the data is presented:

- *Team with highest and lowest not covered practices*

- Teams E5, E6, E8 and E13 have the highest level of not covered practices (53, 54, 54 and 47). Teams E5 and E6 used an adaptive methodology (Scrum); E8 and E13 used a predictive methodology (Waterfall and TSP®).
- E7 is the team with the lowest level of not covered practices (18 practices), it used a hybrid methodology: CMMI®-Scrum.

- *Team with highest and lowest covered processes*
  - Most teams have a better coverage of the Project Management process, as follows: team E1 (77%) used agile practices; followed by teams E7 (54%), E11 (50%) and E12 (50%), 2 of them used a hybrid: CMMI®-Scrum as methodology and 1 used a methodology based on CMMI® model. Finally, teams E2 (35%), E3 (42%), E4 (42%) and E10 (27%); 1 team used a hybrid CMMI®-TSP®, 2 teams used an adaptive methodology (Scrum) and 1 team used a predictive methodology (Waterfall).
  - Only 5 teams have a better coverage of the Software Implementation process, as follows: team E7 (66%) used a hybrid: CMMI®-Scrum as methodology; followed by teams E10(54%) and E11(46%) one used a predictive methodology (Waterfall) and one used a hybrid: CMMI®-Scrum. Finally, teams E12 (37%), and E9 (34%); both teams used a predictive methodology (based on CMMI® and TSP®).

4.3 Middle diagnostic comparison

The middle diagnostic comparison was performed after the execution of the fifth step of the method used and correspond to the milestone 2. Figure 3 summaries the data of practice coverage.

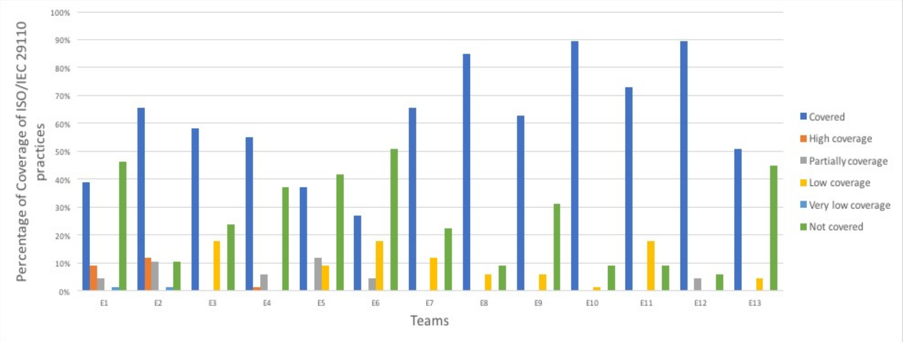

Fig. 3. Milestone 2: middle comparison results after the implementation of the fifth step of the method.

As Figure 3 shows, most of teams have a high and medium percentage of covered practices regarding the Basic profile. If we analyzed the number of covered practices: 4 teams (31%) have from 46 to 67 implemented practices; 8 teams (61%) have from 23 to 45 implemented practices and 1 team (8%) has from 0 to 22 implemented practices.

To understand the performance of the team next data is presented:

- *Team with highest and lowest not covered practices*
  - Teams E12, E11, E10 and E8 had the highest level of covered practices (60, 60, 57 and 49). E12, E10 and E8 used a predictive methodology (methodology based on CMMI® model and waterfall); and E11 used a hybrid methodology: CMMI®-Scrum.
  - E6 is the team with the lowest level of covered practices (18 practices), it used an adaptive methodology (Scrum).
- *Team with highest and lowest covered processes*
  - Most teams have a better coverage of the Project Management process, as follows: team E1 (92%) and E12 (92%) one used agile practices and the other using a predictive methodology (methodology based on CMMI® model); followed by teams E2 (88%) and E10 (88%), E11 (77%) and E9 (73%), E2 used a hybrid: CMMI®-Scrum; E12 used a predictive methodology (methodology based on CMMI® model); E11 used a hybrid methodology: CMMI®-Scrum and E9 used a predictive methodology TSP®. Finally, teams E4 (69%), E3 (65%), E5 (62%), E13 (62%), E7 (62%) and E6 (50%); 4 teams used an adaptive methodology (Scrum); 1 team used a predictive methodology (TSP®) and one team used a hybrid methodology: CMMI®-Scrum.
  - Only 4 teams have a better coverage of Software Implementation process, as follows: teams E8 (98%), E12 (95%), E10 (90%) and E2 (88%); E8 and E10 used a predictive methodology (Waterfall); E12 used a predictive methodology (methodology based on CMMI® model) and E2 used a hybrid methodology CMMI®-TSP®. Finally, E7 (68%) used a hybrid methodology: CMMI®- Scrum.

4.4 Final diagnostic comparison

The final diagnostic comparison was performed after the execution of the sixth step of the method and corresponds to the milestone 3. Figure 4 summaries the data of the total of practices coverage.

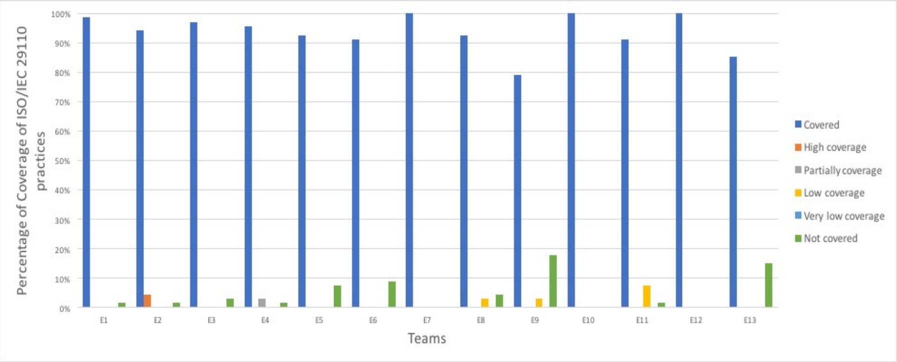

Fig. 4. Milestone 3: final comparison results after the implementation of the sixth step of the followed method.

As Figure 4 shows, all teams have a high percentage of covered practices regarding the Basic profile, having from 53 to 67 implemented practices.

To understand the performance of the team next data is presented:

- *Team with highest and lowest not covered practices*
  - Teams E7, E10, E12 and E1 have the highest level of covered practices (67, 67, 67 and 66). E7 used a hybrid methodology (CMMI®-Scrum); E10 and E12 used a predictive methodology (Waterfall and methodology based on CMMI® model); and E1 uses agile practices.
  - E9 and E13 are the teams with the lowest level of covered practices (53 and 57 practices), both of them used a predictive methodology (TSP®). However, they according to the established coverage levels, they have a high coverage of practices.
- *Team with highest and lowest covered processes*
  - 3 teams have achieved the 100% of coverage of both process (E7, E10 and E12), E7 used a hybrid methodology (CMMI®-Scrum); and E10 and E12 used a predictive methodology (Waterfall and methodology based on CMMI® model). 6 teams have a better coverage of the Project Management process, as follows: teams E1 and E5 (100%), both used an adaptive methodology (agile practices and Scrum); E2, E3, E4 and E13 with (96%), E2 used a hybrid methodology CMMI®-TSP®; E3 and E4 used an adaptive methodology (Scrum); E13 used a predictive methodology (TSP®); and E6, E9 and E11 (92%), E6 used an adaptive methodology (Scrum); E9 used a predictive methodology and E11 used a hybrid methodology (CMMI®-Scrum).
  - Only 4 teams have a better coverage of Software Implementation process, as follows: teams E2 and E4 (100%), E2 used a hybrid methodology (CMMI®-TSP®) and E4 used an adaptive methodology (Scrum). Finally, E1, E3 and E8 (98%), E1 used agile practices; E3 used an adaptive methodology (Scrum) and E8 used a predictive methodology (Waterfall).

5 Discussion

Software standards and models, aimed at the software industry, are developed to contribute to the development of quality products within budget and schedule, by optimizing efforts and resources. However, the implementation of proven practices of these models and standards in software development organizations, especially in VSEs, which must work harder in order to survive, and spend time and effort in improving their operation and processes, is a real challenge.

Previous methods were developed to help in the assessment of ISO/IEC 29110 [13] and to understand the issues that affect its adoption, the needs of VSEs to implement it and their willingness to engage with it [14] have been published.

The method described in this paper produced good results because it facilitated software process improvement activities by considering the values of the SPI Manifesto [15]. The following 3 values of the SPI Manifesto were used when planning and executing the ISO/IEC 29110 improvement activities:

a) People (*"Must Involve people actively and affect their daily activities"):* our method kept the people involved and motivated right from the beginning and throughout all the phases. Besides, our method starts by understanding the culture and the needs of a VSE. Also, all teams collaborated to the selection of ISO/IEC 29110 practices that give value to their processes.
b) Business ("*What one does to make business successful*"): our method motivated VSEs to implement ISO/IEC 29110 activities and tasks by keeping in mind the vision and the business objectives of the VSE. Moreover, the ISO/IEC 29110 is an adaptable standard that can be implemented by VSEs using either the adaptive or the predictive life cycles.
c) Change ("*Process improvement is inherently linked with change*"): our method facilitated the management organizational changes by having each team understanding the ISO/IEC 29110 practices and having them to think about how they will implement the practices to their current processes while they were conducting their software development activities.

6 Conclusions and Future work

This paper presented the results of comparing 13 teams using adaptive and predictive lifecycles, which were executing a method to adapt and implement the software Basic profile of the ISO/IEC 29110 according to their needs. The results showed that, after the last step, of the 6-step method, all teams, having a high-level of coverage of the Project Management and the Software Implementation processes, were ready to initiate a formal audit process toward an ISO/IEC 29110 certification to the software Basic profile.

Each VSE had to invest a minimum effort to attend the training sessions (30 hours), the work meetings (each work meeting took 4 hours) and the milestone meetings (each milestone meeting took 6 hours).

In addition to the meetings held, the effort required by each team to implement the ISO/IEC 29110 varied due to the specific characteristics of each VSE, such as culture, team size, size and type of project, among others.

The ISO/IEC 29110 implementation speed in each VSE was adjusted to meet their needs. The calendar time, between the beginning of the implementation until they were certified, was no longer than 4 months.

Next, the main findings of the use of method as well as the results are listed:

- *Method findings:* (1) with the execution of the steps 1 to 4, it is possible to obtain the commitment of the teams, because they understood the need for improving their processes; (2) the method can be used for both predictive and adaptive methodologies, with hybrid methodologies and even with teams which just followed a set of agile practices.

- *Teams Findings:* (1) at milestone 2, all teams had an important improvement in the covered practices of both processes (see Figure 3) compared with milestone 1 (see Figure 2). Team E2, which used a hybrid approach, i.e. CMMI-TSP, is the team that had the highest improvement of both processes: in project management process from 35% of coverage in the first analysis to 88% of coverage in the middle analysis; and in the software implementation process from 29% to 88%; (2) at milestone 3 (see Figure 4), all teams had a high level of coverage of the practices of the software Basic profile of ISO/IEC 29110 for both processes. However, teams using predictive and hybrid methodologies achieved 100% of coverage (hybrid CMMI-Scrum, waterfall and CMMI); (3) teams using adaptive methodologies implemented more practices related to validation and verification, project monitoring and control, change requests, configuration management and delivery instructions for the project management process and requirements, design, test and traceability for the software implementation process; (4) teams using predictive methodologies implemented more practices related to validation, risk management, change requests and delivery instructions for the project management process and design, test and traceability for the software implementation process.

It is important to mention that we identified that each team had a different level of control and maturity regarding the methodology used to develop software.

Finally, as future work, it would be interesting, amongst other, to measure the percentage of rework before and after the implementation, the level of satisfaction of each VSE about their implementation and the quality of the software delivered (i.e. estimated number of defects remaining after delivery) using the software Basic profile of the ISO/IEC 29110.

References

1. Sanchez-Gordon M-L., de Amescua A., O'Connor R.V., Larrucea X., A standard-based framework to integrate software work in small settings. In computer Standards & Interfaces Volume 54, Part 3, November 2017, pps. 162-17. (2017)
2. Software Engineering Institute. CMMI® for Development, Version 1.3. CMMI-DEV, V1.3. Carnegie Mellon University, Pittsburgh, PA. Version 1.3, CMU/SEI-2010-TR-033, Pittsburgh, PA, November 2010.
3. ISO/IEC/IEEE 12207:2017. Systems and software engineering—Software life cycle processes, International Organization for Standardization (ISO), Geneva, Switzerland, 2017.
4. Laporte, C.Y., Muñoz, M., Mejia Miranda, J., O'Connor, R.V., Applying Software Engineering Standards in Very Small Entities-From Startups to Grownups, IEEE Software, Vol. 35, Issue 1, pp 99-103.
5. ISO/IEC: Software engineering- Lifecycle profiles for Very Small Entities (VSEs) - Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile. ISO/IEC

TR 29110-5-1-2:2011. (2011). Available at no cost from ISO: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

6. Novikov D.A. (2016) Teams: Building, Adaptation and Learning. In: Różewski P., Novikov D., Bakhtadze N., Zaikin O. (eds) New Frontiers in Information and Production Systems Modelling and Analysis. Intelligent Systems Reference Library, vol 98. Springer, Cham
7. Watts, H.: The Team Software Process (TSP^{SM}). Technical report, Carnegie Mellon, Software Engineering Institute CMU/SEI-2000-TR-023 ESC-TR-2000-023, pp. 1–51 (2000)
8. Ahmed, F., Capretz, L. F., Bouktif, S., & Campbell, P. (2015). Soft skills and software development: A reflection from the software industry. arXiv preprint arXiv:1507.06873
9. Laporte C.Y., Muñoz M., Gerançon B., The Education of Students About ISO/IEC 29110 Software Engineering Standards and Their Implementations in Very Small Entities, IEEE Canada International in Humanitarian Technology Conference (IHTC), Toronto, Canada. pps. 94-98. (2017)
10. Laporte C.Y., O'Connor R.: Systems and software engineering standards for very small entities: accomplishments and overview. Computer, IEEE Computer society. 49(8) pps. 84-87 (2016)
11. Project Management Institute. Guide to the Project Management Body of Knowledge, fifth edition. pps. 44-46. ISBN 978-1-935589-67-9. PMI, USA. (2013)
12. Muñoz M., Mejia J., Peña A., Lara G., Laporte, C. Y., Transitioning international software engineering standards to academia: analyzing the results of the adoption of ISO/IEC 29110 in four Mexican universities. Computer Standards & Interfaces. DOI: https://doi.org/10.1016/j.csi.2019.03.008
13. Larrucea, X., Santamaría, I., & Colomo-Palacios, R. (2016). Assessing ISO/IEC29110 by means of ITMark: results from an experience factory. Journal of Software: Evolution and Process, 28(11), 969-980.
14. Sánchez-Gordón, M.L., O'Connor, R.V., & Colomo-Palacios, R. (2015). Evaluating VSEs Viewpoint and Sentiment Towards the ISO/IEC 29110 Standard: A Two Country Grounded Theory Study. In Proceedings of 15th International Conference, SPICE 2015, Communications in Computer and Information Science, Volume 526, 2015, pp. 114-127, Gothenburg, Sweden, June 16-17.
15. Pries-Heje J., and Johansen J., SPI Manifesto. V A.1.2.2010, available at: http://2019.eurospi.net/images/eurospi/DownloadCenter/spi_manifesto.pdf