Teaching and learning strategies for software process subject

José Augusto de Sena Quaresma
Graduate Program in Computer Sciense (PPGCC), Institute of Exact and Natural Sciences (ICEN)
Federal University of Pará (UFPA)
Belém – PA, Brazil
augustoquaresma@ufpa.br

Sandro Ronaldo Bezerra Oliveira
Graduate Program in Computer Sciense (PPGCC), Institute of Exact and Natural Sciences (ICEN)
Federal University of Pará (UFPA)
Belém – PA, Brazil
srbo@ufpa.br

***Abstract*— This Research to Practice Full Paper presents that theoretical subjects are a major concern for researchers in the teaching and learning process, especially in how to improve them so that trained professionals have the necessary skills to act. In the study area of Information Technologies, one of the subjects that is framed in this reality is that of software process. The software production industry considers it essential to increasingly improve its software development processes, so that different models were created for this purpose, such as CMMI (Capability Maturity Model Integration) and MPS.BR (Brazilian Software Process Improvement Model), and norms such as ISO / IEC (International Organization for Standardization / International Electrotechnical Commission), pursuing this objective. In the consultation carried out from a survey of professionals with experience in the software market, it was seen that recent graduates need complementary courses to meet the skills required in the profession. In order to solve this situation, the following research questions were formulated: (i) What strategies can be used to teach the software process? (ii) How to use these strategies in the classroom? The main objective of this work is to describe, from a catalog, how to use the main strategies adopted for teaching and learning the software process. For the development of this research, a literature review was carried out in the main qualified conferences in the area of teaching in Information Technology with the collection of studies dealing with teaching on software process. After that, a description of each teaching strategy was carried out, identifying how to adopt it in the classroom and its example of use in the literature. The strategies covered by the catalog are: Flipped classroom, Project-based learning, Gamification, Serious games, Workshop, Software factory, Playful resources and Scrum Methodology. As a way of evaluating the catalog, a peer review was developed, based on the following steps: (i) planning, where the experts were identified and the creation of the questionnaire was developed; (ii) execution, with consultation with experts in the area of software process and teaching and learning from questionnaires; (iii) analysis and summary, where the experts' information was collected and analyzed and deliberated with due justification as to whether or not they would be implemented in the catalog. The main contribution of this work is a catalog evaluated by experts covering the description of how to adopt the strategies for teaching software process, based on the mapping of qualified literature, as an example of execution.**

Keywords—teaching and learning, software process, strategies.

I. INTRODUCTION

Theoretical subjects are a major concern for researchers in the teaching and learning process, especially in the way they improve so that trained professionals have the necessary skills to act [1]. Within the area of teaching and learning, we can observe several researches to improve this process, through new approaches to knowledge, one of which came into evidence are the student-centered approaches [2, 3, 4].

In Information Technologies one of the subjects that is framed as theoretical is that of software process [1]. In general, we can observe a concern with the trained professionals needed, for not having the necessary skills to work in the job market [5, 6, 7]. In contrast to this reality, the software production industry considers it essential to increasingly improve its software development processes, so much so that models have been created, CMMI (Capability Maturity Model Integration) [8] and MPS.BR (Brazilian Software Process Improvement) [9], and standards such as ISO series (International Organization for Standardization) [10, 11], pursuing this goal.

In view of the research context, the following guiding questions were formulated: (i) What strategies can be used to teach the software process, using centered-student approach? (ii) How to use these strategies in the classroom? Therefore, this work aims to describe, from a catalog, how to use the main strategies adopted for teaching and learning the software process.

The paper consists of: this introductory section, with context, justification, research question and objective, Section 2 with the theoretical background, containing information on teaching and learning and the software process, Section 3 presents some related works to this research, Section 4 presents the literature review used as research methodology, containing the planning, execution and summary of the results, Section 5 presents a catalog of teaching strategies in the software process, containing discussions about the collection of literature in the formation of a catalog to assist the teacher, Section 6 presents the evaluation of this catalog by the peer review technique, and Section 7 presents the conclusion and future works.

II. BACKGROUND

This section presents the theoretical background necessary for a good understanding of this paper.

A. Teaching and Learning

Teaching and learning is a discussion of great importance for the development of students in skills necessary for the job market [1]. Over the years, several approaches have been proposed to deal with the theme [12]. One of the approaches that gained prominence due to the number of strategies linked to it is the student-centered approach [12].

The student-centered teaching methodology allowed a change to the paradigm already established in education [2].

This is due to the fact that the traditional study, that is, centered on the teacher, had as objective only the transfer of knowledge [3]. The student-centered approach, on the other hand, makes it possible for the student to play a leading role in the teaching and learning process, as well as valuing educational, professional and personal knowledge and growth [2, 3]. This reality does not exempt the teacher from the responsibility, which remains a fundamental part, since it guides the student towards the construction of knowledge [2, 3].

The student-centered approach envisages that it will be the center of the teaching and learning process. Some of these strategies will be present in the catalog, such as: flipped classroom [3], project-based learning [2], gamification [24], serious games [4, 29], workshop [12], software factory [33], playful resources [34] and scrum methodology [35].

B. Software Process

According to the SWEBOK (Software Engineering Body of Knowledge) guide [13], the software process is conceptualized as a set of interrelated activities and tasks that transform input work products into output work products. To define the software process, the same guide will inform it of the need for necessary inputs, transforming work activities and generated outputs. It is worth noting that activities can be broken down into smaller tasks.

According to [13], a software process can include subprocesses, that is, in the development of a software, have a subprocess specified only on requirements collection and validation, or else a subprocess only for software validation and verification. Another point to be dealt with about the process is its notation, which according to the authors are through may include textual lists of activities and constituent tasks described in natural language, data flow diagrams, BPMN (Business Process Modeling Notation) and activity diagrams in UML (Unified Modeling Language).

III. Related Works

Teaching and learning within the area of software engineering has become evident. In this aspect, we have the development of some works [14, 15] that report the development of catalogs for use in the teaching and learning process.

In [14] there is the development of a teaching catalog through educational software for computer programming, which was generated through systematic mapping. In our analysis, we observed the restriction of the mapping performed because it adopts educational software as a reference only.

In the work of [15], it was developed a literature review on game evaluation with regard to software engineering and learning. In this study, the content worked on in these games is identified, which is software project management. In our analysis, we observed the review restriction only for games, as it could work with other forms of learning evaluation.

Therefore, within the area of software engineering with regard to software process, we do not yet have a catalog with student-centered teaching strategies as to assist teachers in this subject.

IV. Literature Review

Literature review is the means by which the researcher can identify the scientific knowledge existing in a given area, in order to plan his research, avoiding the duplication and repetition of past mistakes [16].

The chosen review format is ad-hoc, as it allows it to be performed freely, but with established criteria, they are: (i) content, which is teaching, software process and quality, (ii) time, which is 5 years from 2015 to 2019, (iii) search venue, which is qualified conferences for computer science, which are in the area of teaching, education and computer science.

A. Planning

In planning the review, the first item to be considered is the research question is identified as one of the ways to guide the review work to identify in the literature whether there are already described approaches, or the possible gap's to conduct research. Therefore, for this work we have the following research question: (QP01) What are the main approaches to support the teaching and learning process or software quality in the context of undergraduate and graduate courses in Information Technology, which are centered in the student and with industry training practices?

Another point to be considered is the identification of the search sources and the scope of the research. In this sense, the research was carried out manually in the main conferences on Software Engineering, Teaching and Education, which are qualified for Computer Science. The choice for the general scope of Software Engineering is because, as indicated in the ACM / IEEE (Association for Computing Machinery / Institute of Electrical and Electronics Engineers) curriculum [17] and the base document for Information Technology courses at SBC (Brazilian Computer Society) [18] is the general area on subjects related to the contents on Software Process and Quality.

During the collection process of primary studies, access via a Federal University in Brazil was adopted, through the periodical CAPES (Coordination for the Improvement of Higher Education Personnel) in Brazil, which allows access to published and indexed papers on the conferences. The selected conferences for computer science are available in Table I.

B. Execution

The review was carried out as follows: (i) each of the conference proceedings was consulted from 2015 to 2019 for the inclusion of paper, (ii) adherence to the research question was visualized, that is, it needs to be an approach to teaching software process or quality, and (iii) during the search, titles, abstracts and, if necessary, even parts of the study were read as a way to identify their inclusion in the research.

C. Summary of results

For the extraction and synthesis of the data, an evaluation model was adopted for each of the included papers, according to the following topics: strengths, weaknesses, limitations, methodology adopted, the contribution and future work. In the analysis of the points extracted from the papers, the synthesis and thematic analysis proposed by [19] was adopted, where the themes associated with each topic where the paper was evaluated were identified. This study selected a total of 29 papers relevant to the research question,

as shown in Table II. It is noteworthy that the CSEE&T conference did not take place in 2018, so the table shows "-" instead of "0".

TABLE I. CONFERENCES USED IN THE AD-HOC REVIEW.

Acronym	Name
ICSE	International Conference on Software Engineering
CSEE&T	Conference on Software Engineering Education and Training
ITICSE	Conference on Innovation and Technology in Computer Science Education
CSEDU	International Conference on Computer Supported Education
SIGCSE	ACM Technical Symposium on Computer Science Education
ICSOFT	International Conference on Software and Data Technologies
SBIE	Brazilian Symposium on Informatics in Education
SBQS	Brazilian Symposium on Software Quality
SBES	Brazilian Symposium on Software Engineering
FIE	IEEE Frontiers in Education Conference
WEI	Workshop on Computer Education

TABLE II. INCLUSION OF STUDIES PER YEAR.

Acronym	2015	2016	2017	2018	2019	Total
ICSE	0	2	0	0	0	2
CSEE&T	0	0	1	-	0	1
ITICSE	0	0	0	0	0	0
CSEDU	0	0	0	0	0	0
SIGCSE	0	0	0	0	1	1
ICSOFT	0	0	0	0	0	0
SBIE	1	0	2	0	0	3
SBQS	1	0	0	4	2	7
SBES	2	0	2	1	2	7
FIE	0	2	1	0	1	4
WEI	0	1	0	0	3	4

As the original objective of the article is not to present the literature review with its results, this information can be consulted in the research by [20].

V. A CATALOG OF TEACHING STRATEGIES ON SOFTWARE PROCESS

This section describes the catalog with strategies for teaching and learning in the software process.

A. Catalog Identification

The identification of the catalog informs its authors and that it is an integral part of a doctoral research underway at a Brazilian Federal University. In addition, it indicates that its constitution was carried out through a literature review.

B. Description of Catalog Sections

The catalog consists of teaching strategies divided into sections. Within each specific section is the necessary information on each of the teaching strategies, as well as: a brief description, how to use it?, which indicates the methodological procedures, necessary infrastructure and the evaluation process, and examples in the literature.

1) Flipped Classroom: The flipped classroom is the inversion of events and roles inside and outside the classroom [3]. In this teaching strategy, the teacher has the role of mediator of knowledge, while the student has the role of propagator of knowledge. Within the classroom, debates, simulation, group work, problem solving are stimulated, so we consider it as the active student for the development of its activities and tasks. In the external environment of the classroom, the student is invited to seek alternative knowledge materials on the subject as a way to stimulate his ability to debate.

a) Description: Baker initially coined the active methodology of the flipped classroom in the literature through the term "Flipped Classroom" at an international conference on education. However, the origin of the approach dates back to 1991 adopted by Eric Mazur, which was later published in 1997 through the book "Peer instruction: a User's Manual". In this first version of the approach, we had the concept of students carrying out a prior study of the material in order to make conceptual tests in class.

b) How to use it?: The methodological procedures within the flipped classroom are: (i) teacher prepares the base materials, (ii) students access the content before class, (iii) in class, the division of themes by group, (iv) in class a discussion section on the main points and clarification of doubts, (v) outside the classroom, students prepare presentations with the basic contents and new materials researched, and (vi) in the classroom, the groups present the contents and deliver a written report of the contents adopted.

The necessary infrastructures for using the flipped classroom are: brush and blackboard, multimedia resource, such as presentation projector and an environment that can be an extension to the classroom, as a way to provide the necessary content.

The evaluation process will take place in three steps: (i) participation in the initial debate class on the theme, carried out by the teacher, (ii) peer review carried out by classmates with pre-established criteria, between teachers and students, and questions to colleagues about the content presented, and (iii) evaluation of the knowledge produced reports that accompany the presentation sent by the student, made by the teacher.

c) Examples: The examples collected in the literature review are [21, 22], and a better description can be found in [20].

2) Project-Based Learning: Project-Based Learning (PBL) is an active and collaborative methodology in which students must form groups, which are called research or investigation tasks [2]. The purpose of this type of methodology is to stimulate students' critical thinking, leading them to collect information, formulate and refine questions, make predictions and share their ideas and conclusions with colleagues. In addition to being a teaching

strategy that can increase students' motivation, contributing to the development of skills and promoting learning centered on the figure of the student.

a) Description: Project-based learning considers the profile of the teacher as a mediator of knowledge, where he is to assist in the structure and guide in the development of the project, but does not do the project. The student has an active profile, as he will develop the activity, identifying points for improvement and discussing the evolution with his colleagues.

b) How to use it?: The methodological procedures within project-based learning are: (i) the teacher prepares the base materials, (ii) students access the content before class, (iii) in class, the division of themes by group, (iv) starting of the design of the project still in the classroom, with questions section, (v) outside the classroom students develop the project with a report on the results achieved, and (vi) in the classroom, the groups present the results obtained with the completion of the project and learning lessons.

The necessary infrastructure for using the PBL are: brush and blackboard, multimedia resource, such as presentation projector and an environment that can be an extension to the classroom, as a way to provide the necessary content.

The evaluation process will take place in three steps: (i) presentation of the project's results in class, (ii) peer review carried out by classmates with the pre-established criteria with the teacher on the presentation of the project results, and (iii) evaluation of results reports by the teacher.

c) Examples: The examples collected in the literature are the studies of [23, 24], and a better discussion can be found in [20].

3) Gamification: Gamification is the use of game elements in non-game contexts [25]. In general, adopting gamification encourages competition, collaboration and motivation of students in the teaching and learning process.

a) Description: The game elements allow for a more stimulating classroom environment for the students' learning process. For the content of the software process, we recommend adopting the following game elements:

(i) The rules of the game, right at the beginning of the subject there will be a class presenting the elements composed of this teaching plan and how the student can be approved or failed within the subject,

(ii) The epic narrative in the theoretical class with concept maps, identifying and contextualizing the importance of that content for the student within the course and in the job market,

(iii) The score, which corresponds to activities performed correctly within the subject the student will receive this element,

(iv) Ranking, a visualization available to the student of how they are doing within the subject and in relation to their colleagues,

(v) Use of an avatar in the personification of the flipped classroom,

(vi) Feedback, delivery to students of their progress within the subject, as well as verification by the teacher about the strategies adopted.

b) How to use It?: The methodological procedures for the development of gamification will be: (i) development of the initial class with all the rules of the game, (ii) use of the epic narrative in the theoretical and practical class, (iii) definition of the score for each activity developed within the subject, (iv) programming of the electronic spreadsheet to generate the class ranking automatically, (v) availability of the flipped classroom for the use of an avatar, (vi) availability of the ranking and the correction map of activities via the flipped classroom, and (vii) individual submission of the correction for the scored activities performed.

The necessary infrastructure for the development of gamification is the flipped classroom, a device with an internet connection and document software and spreadsheet.

The evaluation process of gamification runs through all other teaching strategies adopted in the classroom, since the grade given to the student will be the accumulation of points within the gamification.

c) Examples: The examples collected in the literature are the studies of [24, 26, 27, 28], and a better discussion can be found in [20].

4) Serious Game: Serious games in the literature as a serious game can be defined as a game with purpose. A game has the following characteristics: goals, feedback system rules and voluntary participation.

a) Description: The difference in gamification for serious games is their range and format. In games, students need to have a clear goal, the rules that will be supporting them in the classroom, constant feedback on their evolution within the game and have their voluntary participation. Within gamification it can choose game elements, which can be the characteristics and adopt them in the classroom.

b) How to use it?: The methodological procedures for the adoption of serious games in the classroom are: (i) define the goal to be achieved in the game, (ii) define the content which will be treated in the game, (iii) define which rules will make the player reach that goal, (iv) definition of the feedback system, verifying how to present to the students their performance within the game, (v) check points that allow the game to be playful within the classroom, (vi) use the game developed in the classroom environment, and (vii) conduct an evaluation to see how the students' learning was.

The necessary infrastructure for the development of the game depends on its planning and how it will be executed, for example: (i) for electronic games, students will have to have access to devices, (ii) for instructional games, only the classroom environment is sufficient.

For games in the classroom, it is suggested not to imply evaluative grades since they should show this playful experience. However, in order to evaluate students' knowledge, it is worth applying knowledge tests after the moment of playing in the classroom.

c) Examples: The examples collected in the literature are the studies of [4, 29, 30, 31, 32], and a better discussion can be found in [20].

5) Workshop: A workshop is an intensive course or seminar of short duration, where technical procedures and forms of use, are explained and applied.

a) Description: The use of the workshop for education is a way to make knowledge practical, that is, to use that knowledge in a controlled classroom environment.

b) How to use it?: The methodological procedures necessary for the implementation of the workshop in the classroom are: (i) definition of the content to be worked on, (ii) check how it can be worked in a practical way, either in the literature or through the knowledge arising from the labor market, (iii) identify the best way to work it in the class, (iv) present the content in a practical way, and (v) asking students to execute the content in a practical way, following the teacher's examples.

The necessary infrastructure for the development of this teaching strategy depends on the adopted planning, we have as examples: oral presentation of a content, in this situation the infrastructure already present in the classroom supplies the need.

The evaluation of the workshop is the analysis of the execution performed by the student of the technique or knowledge taught in a practical way inside the classroom.

c) Examples: The example collected in the literature is the study of [12], and a better discussion can be found in [20].

6) Software Factory: A software factory within an academic environment is a way to provide students with an experience of the labor market within the academy.

a) Description: The software factory within the teaching of software process is the adoption of a software process organization within a classroom, with the definition of the roles of the team, how the software development process will be modeled, a quality model or standard, how to meet growing customer demands. Therefore, the software factory is configured as a simulation of a corporate environment within the classroom.

b) How to use it?: The ideal form for the development of a software factory is: (i) definition by the teacher of the scope and goal required for the factory, (ii) development of a real software development context, which the students had to perform, (iii) delivery of the context as well as the scope of the factory and its goal for the student team, as well as solving doubts and scheduling development milestones, (iv) checking the progress of the project through scheduling the milestones, and (v) delivery of the presentation software to the class.

The necessary infrastructure for the software factory are: (i) notebooks, for the development of software and work prodcuts, (ii) blackboard and brush, for the teacher to present the hypothetical situations.

Evaluation within the software factory is how students perform their role within the process, as well as how they will define the software role and process collectively.

c) Examples: The example collected in the literature is the study of [33], and a better discussion can be found in [20].

7) Playful Resources: Playful resources are elements that will bring fun to students while learning about certain content.

a) Description: The use of playful resources in the classroom allows the formality imposed by the environment to be more relaxed, such a situation is necessary for the student to be more open to the learning process.

b) How to use it?: The methodological procedures for using playful resources in the classroom are: (i) definition of the content to be addressed, (ii) identification of the objective to be achieved when using that resource, (iii) planning a dynamic, which allows fun among students, and (iv) adoption of dynamics in the classroom.

The necessary infrastructure for the development of the resource depends on the planning of the activity adopted for the classroom.

For this teaching strategy, as students are expected to participate more without imposing a grade, a score for participation in the activity is appropriate.

c) Examples: The example collected in the literature is the study of [34], and a better discussion can be found in [20].

8) Scrum Methodology: Jeff Sutherland created The Scrum methodology in 1993 observing the writings of Takeuchi and Nonaka, where a comparison of the high performance teams was made to the existing “Scrum” formation in the Rugby teams.

a) Description: The Scrum methodology is a Lean approach to project management, mainly in software development. Scrum is a framework that is classified as an agile methodology.

b) How to use it?: The methodological procedures for the development of the scrum methodology in the classroom are: (i) the teacher describes the context in which the project will be developed, as well as necessary requirements, (ii) definition of the roles of the methodology, (iii) initial meeting with the product owner, (iv) sprint planning meeting, (v) daily meeting of 10 minutes to check the development of the project, (vi) sprint review meeting identifying which items were delivered and which were not possible, and (vii) return to step iv until the product is fully delivered to the customer.

The necessary infrastructure for the development of the Scrum methodology is: (i) notebooks for software programming, (ii) framework to simulate the backlog, and (iii) letters to execute the planning pocker.

The evaluation of this methodology will be based on the student's role in which he was involved.

c) Examples: The example collected in the literature is the study of [35], and a better discussion can be found in [20].

C. How to use the Catalog?

The instructions for using the catalog are so that teachers can make better the use of this. Therefore, we indicate if they need the full use of the strategy, since it has its description, the methodological procedures, infrastructure and evaluation process, as well as at least one example in the literature to take as a basis.

VI. Evaluation of the Catalog

The strategy catalog was evaluated using the peer review technique. Peer review can be considered as the most effective and efficient mechanism to guarantee the quality, reliability, integrity and consistency of academic literature [36]. Information on using this procedure will be detailed in the following subsections.

A. Planning

The planning of the peer review took place in two steps: (i) definition of the data collection instrument, where it is possible to ascertain the expertise of the consulted professional in relation to student-centered approaches and software process, as well as the format adopted will be described for the evaluation of the teaching plan, and (ii) research and choice of professionals to carry out the peer review in the catalog.

B. Execution

The review was carried out using the following steps: (i) consultation with the professional if he / she agrees to participate in the peer review, (ii) submission of the evaluation form as well as a strategy catalog, (iii) presentation of the catalog for the expert, made by the researcher, (iv) completion of the instrument and indication of the necessary changes in the strategy catalog, and (v) analysis of peer statements about the document.

C. Abnalysis and Summary

The analysis of the peer review begins with the identification of the participant's profile. This first part of the instrument collected information such as software process teaching in undergraduate courses, use of student-centered approaches within the classroom, which approaches were taken when teaching. With this information, it is possible to state that the participant is an expert both in the software process and in student-centered approaches.

The review form had the second section with instructions for evaluating the strategy catalog, where the pair indicated: (i) the category, identified in the instrument, which could be high technician, low technician, editorial, questioning and general, (ii) the catalog item with any problem, which is the location within the catalog that needs revision or change, (iii) justified comment on the need for adjustment, and (iv) suggestion, a form of adjustment indicated by the expert. This section generated Table III.

Table III identifies the revisions made by the expert in the strategy catalog, as a way of improving all the indications were made according to the suggestions sent by the expert.

VII. Conclusion and Future Works

The software process subject is framed as theoretical. However, in general we have trained professionals who do not meet the profile of skills in the job market, as we identified in the search for information through the literature. On the other hand, the software industries increasingly seek to specialize in the development of processes through quality models and standards for the software process.

This work aimed to describe, from a catalog, how to use the main strategies adopted for teaching and learning the software process. We achieve this goal by conducting an almost review of the literature and describe 8 teaching strategies with student-centered approaches and examples of use within the literature.

TABLE III. Item to be solved after Peer Review

Category	Item	Comment with Justification	Suggestion
Low Technician	Description of Teaching Strategies	The teaching strategies were not well detailed.	Presenting the teaching strategies in detail by reviewing their methodological procedures, infrastructure and evaluation.
Low Technician	Example of Use of Teaching Strategies	There is no reference to studies found in the specialized literature to be used as examples of use.	Refering to studies that deals with the literature review used to understand the references used as examples to compose teaching strategies.

Therefore, through the 8 teaching strategies we were able to answer the first research question (i) about which strategies we can adopt to teach software process, with a student-centered approach. To answer the second research question (ii), which is how to use these strategies in the classroom, we have the formulation of the catalog with an indication of conceptualization, how to use it and an example from the literature.

Future work after making the catalog is the development of teaching plans for the software process subject, adopting the strategies collected and expanding the collection universe for journals and conferences in the computer science.

References

[1] R. Prikladnicki, A. B. Albuquerque, C. G. Von Wangenheim, and R. Cabral, “Software engineering teaching: challenges, teaching strategies and lessons learned”. FEES-Fórum de Educação em Engenharia de Software, 2009. 1-8.

[2] W. N. Bender, “Project-based learning: differentiated education for the 21st century”. Penso Editora, 2015.

[3] L. A. Schneiders, “The flipped classroom method”. Lajeado: Ed. da Univates. 2018.

[4] G. G. Moreira and A. B. S. Marques, “Evaluating the students' experience with the Scrum Card Game: an experience report in a Software Engineering course”. In Proceedings of the 17th Brazilian Symposium on Software Quality (pp. 344-353). October 2018.

[5] S. Meira, “Information Systems and Software Engineering - Where's the Schools?”. Revista da SBC Engenharia de Software - Qual é o impacto da ES no mercado de Computação e na sociedade como um todo? Porto Alegre, Brasil. 2015.

[6] A. M. Moreno, M. I. Sanchez-Segura, F. Medina-Dominguez, and L. Carvajal, “Balancing software engineering education and industrial needs”. Journal of systems and software, 85(7), 1607-1620. 2012.

[7] J. A. S. Quaresma and S. R. B. Oliveira, “A Study on the perception of the teaching learning of software process in the academia and industry: a survey application”. In: 18th CONTECSI. 2021.

[8] SEI, “CMMI® for Development, Version 1.3: Improving processes for developing better products and services”. USA, 2010.

[9] SOFTEX, “MPS.BR - Brazilian Software Process Improvement: MPS Software General Guide”. Brazil, 2021.

[10] ISO/IEC, “ISO/IEC 12207 Systems and software engineering– Software life cycle processes”. Geneve: ISO, 2017

[11] ISO/IEC, “ISO/IEC 33020 – Information Technology – Process Assessment – Process Measurement Framework for Assessment of Process Capability”. Geneve: ISO, 2019.

[12] C. Portela, A. Vasconcelos, S. Oliveira, and M. Souza, “The use of industry training strategies in a software engineering course: an experience report”. IEEE 30th Conference on Software Engineering Education and Training (CSEE&T). 2017.

[13] P. Bourque and R. Fairley, “Software engineering body of knowledge (SWEBOK) v3”. IEEE Computer Society, EUA. 2014.

[14] A. Marcolino and E. F. Barbosa, “Educational software for teaching programming: A systematic mapping”. In Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE) (Vol. 26, No. 1, p. 190). 2015.

[15] C. G. Von Wangenheim. D. Kochanski, and R. Savi, “Systematic review on game evaluation for software engineering learning in Brazil”. Proceedings of the FEES-Fórum de Educação em Engenharia de Software. 2009.

[16] S. N. Mafra and G. H. Travassos, “Primary and Secondary Studies supporting the search for Evidence in Software Engineering”. Relatório Técnico, RT-ES, 687(06). 2006.

[17] M. Sahami, S. Roach, E. Cuadros-Vargas, and R. LeBlanc, “ACM/IEEE-CS computer science curriculum 2013: reviewing the ironman report”. In Proceeding of the 44th ACM technical symposium on Computer science education(pp. 13-14). March 2013.

[18] Sociedade Brasileira da Computação, “Training references for undergraduate courses in Computing 2017”. Brasil. 2017.

[19] D. S. Cruzes and R. Dyba, “Recommended steps for thematic synthesis in software engineering”. In 2011 international symposium on empirical software engineering and measurement (pp. 275-284). IEEE. 2011.

[20] J. A. S. Quaresma and S. R. B. Oliveira, “Approaches for teaching and learning in the discipline Algorithms: A systematic review of the literature”. In: 17th CONTECSI. 2020.

[21] L. N. Paschoal, L. Silva, and S. Souza, “Flipped classroom approach compared to the traditional teaching model: an empirical investigation in the context of software testing”. In Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE) (Vol. 28, No. 1, p. 476). 2017.

[22] L. N. Paschoal, B. R. Oliveira, E. Y. Nakagawa, and S. R. Souza, “Can we use the Flipped Classroom Model to teach Black-box Testing to Computer Students?”. In Proceedings of the XVIII Brazilian Symposium on Software Quality (pp. 158-167). 2019.

[23] M. Souza, R. Moreira, and E. Figueiredo, “Students Perception on the use of Project-Based Learning in Software Engineering Education”. In Proceedings of the XXXIII Brazilian Symposium on Software Engineering (pp. 537-546). 2019.

[24] M. Souza, R. Moreira, and E. Figueiredo, “Playing the Project: Incorporating Gamification into Project-based Approaches for Software Engineering Education”. In Anais do XXVII Workshop sobre Educação em Computação (pp. 71-80). SBC. 2019.

[25] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, “From game design elements to gamefulness: defining" gamification". In Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments (pp. 9-15). 2011.

[26] E. D. Santos and S. R. B. Oliveira, “Gamification and Evaluation the Use of the Function Points Analysis Technique in Software Quality Subjects: The Experimental Studies”. In Proceedings of the 17th Brazilian Symposium on Software Quality (pp. 354-362). 2018.

[27] G. M. Jesus, L. N. Paschoal, F. C. Ferrari, and S. R. Souza, “Is It Worth Using Gamification on Software Testing Education? An Experience Report”. In Proceedings of the XVIII Brazilian Symposium on Software Quality (pp. 178-187). 2019.

[28] F. Pinto and P. Silva, “EduGamification: a gamification methodology to support the teaching-learning process”. In Anais do XXVII Workshop sobre Educação em Computação (pp. 414-428). SBC. 2019.

[29] V. Moura and G. Santos, “ProcSoft: A Board Game to Teach Software Processes Based on ISO/IEC 29110 Standard”. In Proceedings of the 17th Brazilian Symposium on Software Quality (pp. 363-372). 2018.

[30] T. A. Beppe, Í L. Araújo, B. S. Aragão, I. S. Santos, D. Ximenes, and R. M. C. Andrade, “GreaTest: a card game to motivate the software testing learning”. In Proceedings of the XXXII Brazilian Symposium on Software Engineering (pp. 298-307). 2018.

[31] B. R. Maxim, R. Kaur, C. Apzynski, D. Edwards, and E. Evans, “An agile software engineering process improvement game”. In 2016 IEEE Frontiers in education Conference (FIE) (pp. 1-4). IEEE. 2016.

[32] P. H. D. Valle, A. M. Toda, E. F. Barbosa, and J. C. Maldonado, “Educational games: A contribution to software testing education”. In 2017 IEEE Frontiers in education Conference (FIE) (pp. 1-8). IEEE. 2017.

[33] V. C. B. Pocivi, W. F. Marins, and K. S. Diogo, “Turing Technology Factory: A Strategy for Effective Teaching of Software Engineering”. In Proceedings of the XXXIII Brazilian Symposium on Software Engineering (pp. 523-531). 2019.

[34] R. Castro and G. Souza, “The Use of Playful Resources for Teaching Process in Software Engineering”. In Anais do XXIV Workshop sobre Educação em Computação (pp. 270-279). SBC. 2020.

[35] F. G. Rocha, R. F. Sabino, and R. H. L. Acipreste, “The SCRUM methodology as a mobilizer of pedagogical practice: a look at software engineering”. Fórum de educaçao em engenharia de software, 13-23. 2015.

[36] A. Davyt and L. Velho, “Science evaluation and peer review: past and present. What will the future look like?”. História, Ciências, Saúde-Manguinhos, 7(1), 93-116. 2000.