An Experimental Evaluation of a Teaching Approach for Statistical Process Control in Computer Courses

Julio Furtado, Federal University of Pará (UFPA), Brazil

Sandro Ronaldo Bezerra Oliveira, Federal University of Pará (UFPA), Brazil

Rafael Oliveira Chaves, Federal University of Pará (UFPA), Brazil

 https://orcid.org/0000-0002-6333-8776

Adeildo Telles, Federal University of Amapá (UNIFAP), Brazil

Adolfo Colares, Federal University of Amapá (UNIFAP), Brazil

ABSTRACT

In organizations that are seeking a high degree of maturity, it is necessary to achieve a statistical control of software processes and to know their behavior and operational performance. The approach adopted for the research involves reading articles and experience performance reports, practical cases, discussion, the use of games and simulators, practical projects, and reflection by students on the knowledge learned and activities carried out. The evaluation was conducted with undergraduates enrolled in a Computer Science Bachelor's degree programme, who were divided into a control group and an experimental group. At the end, the two groups carried out a practical project to evaluate the learning effectiveness reached by the students. The results of the study suggest that there was a difference in the effectiveness of the learning resulting from the teaching approach and traditional instruction. The authors observed a mean gain of 30.06% in the experimental group, which is evidence of this rise in the learning effectiveness.

KEYWORDS

INTRODUCTION

A Statistical Process Control (SPC) must be stable and repeatable and includes a set of techniques for achieving this goal. For some time, it has been employed for continuous process improvement in industry in general, but with regard to software organizations, its use can be regarded as relatively recent (Alhassan and Jawawi, 2014).

The driving force behind the need to find a better way to teach SPC topics to software engineers has arisen from empirical considerations as well as through the observations and consulting of the authors during the process improvement. The Brazilian software industry has widely adopted the use

of quality models, such as Capability Maturity Model Integration for Development - CMMI-DEV (SEI, 2010) and the Brazilian Reference Model for Software Process Improvement - MR-MPS-SW (SOFTEX, 2016), which normally takes place through the Process Improvement Software (SPI) programmes implemented by consulting firms.

As consultants, we adopt several training strategies for the transfer of knowledge needed for the specific practices or expected results included in the process area, particularly in SPC, in order to develop the skills and competencies required by the technical team of the organization.

In the light of this, this study designed a teaching methodology for SPC. It should be noted that our objective is not to prepare an undergraduate program in Computing, but rather, to improve the teaching of SPC as a training scheme that can be administered at any level of education or in any work environment, both in the academic world and industry. This methodology is composed by a complete syllabus and a set of strategies for teaching this syllabus.

A formal experiment was conducted with undergraduates enrolled in a Computer Science Bachelor's degree program to compare the learning effectiveness of the new teaching approach with that of a class taught by traditional lecturers. This decision was made because the researchers had ease of access to this student population. The results of the study showed that there was a significant difference between the degrees of learning effectiveness, since the new teaching approach achieved better results than the use of traditional instruction.

THE BACKGROUND TO STATISTICAL PROCESS CONTROL

The quest for quality improvement must follow a continuous pattern and be based on the constant application of strategies that allow problems to be recognised, and certain activities to be given priority, while adopting a proactive stance.

With regard to the maturity model of the CMMI-DEV process, it is clear that a distinction should be made that depends on the level of maturity of a software organization (Rocha, Souza and Barcellos, 2009). Until Level 3 of CMMI-DEV, a traditional analysis is conducted, which involves collecting the data of a project and comparing them with the estimated values. In contrast, at the higher levels of the model (CMMI-DEV), it is necessary to statistically control the software process to determine its organizational objectives and its clients, as well as to provide opportunities for taking new measures.

SPC then draws on the data collected in the projects, together with control charts, to analyse the behaviour of the selected processes, which are of critical importance to enable the organization to achieve its business objectives (Paranthaman, 1990). Thus, the aim of SPC is to reduce the variability of the processes and improve the quality of the production results. At the same time, its main purpose is to ensure stable processes, that is, processes that behave in a predictable and repeatable way.

The effectiveness of SPC is based on the principle that if a process occurs under certain conditions, this process will only be subject to the effects of Common Causes. Thus, if a process is known, it is possible to predict its exact behaviour in future executions (Pinton, 1997).

Any processes that are considered to be unstable, and which do not behave in a predictable and repeatable way, should have their causes of instability investigated and corrected to ensure the stabilization of the process. Once the process has been finally stabilized, measures can be taken to improve its capacity for action with the aim of achieving continuous process improvement.

TEACHING STATISTICAL PROCESS CONTROL TO SOFTWARE ENGINEERS

According to (Nunes *et al.*, 2016), it is first necessary to determine what professional competences will be required and then define the means that can enable the students to develop them.

A survey was carried out with software engineers to understand the real needs of the software industry for SPC, and with the aim of determining which SPC topics were the most suitable for their

Table 1. Syllabus units

Unit	Objective	Related Basic Skill
Business processes and objectives	Required to teach the basic concepts of process and organization, teaching analytical methods, modeling; the implementation of processes and the definition of critical processes	1, 2, 6
Measurement	This aims at teaching how to define and design a measurement plan	3, 4, 5, 7, 8
Statistical control	The importance of SPC will be taught, as well as assistance given on how to use control charts and how to carry out "cause and effect" assessment	9, 10
Evaluation of capacity and process improvement	This will teach baseline concepts and process capability, as well as how to improve processes	11, 12, 13

organizations (Furtado and Oliveira, 2017). After this, it was possible to classify the 13 basic skills that are needed for a software engineer to work in SPC. These can be listed as follows:

1. Finding out which processes are aligned with the quantitative objectives of business;
2. Determining what processes are needed to provide the information necessary to achieve the organization's business objectives;
3. An ability to define the aims of process measurement;
4. Establishing the measurable relationships between the process features that contribute to the process performance;
5. Defining the quantitative objectives for process quality management and performance that are aligned to the need for information and the ability to achieve business objectives;
6. Selecting the processes that will be the object of a performance analysis;
7. The ability to take appropriate measures for the process performance analysis;
8. Collecting, validating, and communicating the results of measurements to monitor to what extent the quantitative targets for process performance have been achieved;
9. Selecting the techniques needed to analyse the data collected;
10. Analysing the measurement data with regard to the special causes of process variation;
11. Characterizing process performance;
12. Taking corrective measures to address the special causes of variation;
13. Establishing, improving and adjusting the process performance models.

Subject, Syllabus and Goals

Once these skills had been determined, it was possible to set out the syllabus that could cover the whole of this background. Thus, the subject-area was divided into 4 units, as can be seen in Table 1:

The expected level of cognitive ability for each unit and its content was defined by means of terminology based on Bloom's taxonomy (Bloom, 1956). This consists of *remembering*, *understanding* and *application* categories, which can be defined as follows: a) *remembering*, i.e. memorizing the material previously taught; b) *understanding*, i.e. understanding the information and meaning of the material taught; and c) *application*, i.e. using the material learned in new and concrete situations. It should be emphasized that the Application category includes Understanding, and this includes Remembering (Nunes *et al.*, 2016).

Table 2 provides a summary of the syllabus, as well as the expected results and expected learning levels for each topic.

Table 2. Subject syllabus and goals

| Topics	Expected Results	Level
1.1 Introduction to processes	The student must know the basic concepts and representation of processes.	Remember
	The student must be able to see the relationship between the quality of the process and the quality of the product.	Remember
1.2 Processes and organizational structure	The student must be able to see the relationship between the process and the organizational structure.	Remember
1.3 Definition and implementation of processes	The student must understand the analysis and modelling of processes.	Understand
	The student must know the management of the processes implementation.	Remember
1.4 Decision-making process	The student must understand the decision-making process.	Understand
1.5 Critical processes for the business	The student must be able to identify and select (under supervision) the critical processes of an organization.	Apply
2.1 Measurement concepts	The student must know the basics of software metering.	Remember
	The student must be able to understand how measurement objectives should support the organization's objectives.	Remember
2.2 Measurement process	The student must be able to define and execute (under supervision) a measurement plan.	Understand and Apply
3.1 Introduction to Statistical Control	The student must be aware of the importance of statistical control.	Remember
3.2 Control charts	The student must understand the various types of control charts.	Understand
	The student must be able to select the control charts that best suit a situation.	Apply
3.3 Cause and effect assessment	The student must be able to evaluate the measurement data and identify the special causes of process variation	Understand and Apply
4.1 Assessment of process capability	The student must be able to characterize the performance of a process.	Apply
	The student must be able to establish performance models for the process.	Apply
4.2 Improve of process performance	The student must be able to propose adjustments and improvements to the process performance models	Apply

A complete syllabus of the subject, together with the expectations regarding the students' results, can be consulted in Furtado and Oliveira (2018a).

THE TEACHING APPROACH

In the case of ACM/IEEE (2013), Software Engineering (SE) is a subject that is concerned with applying theory, knowledge and practice to the effective and efficient development of software systems that meet the requirements of users. This requires going beyond the traditional way of teaching, paying more attention to the complexity of social interactions, discussing how collaborative software development occurs in a real-world environment (Pinto, Steinmacher and Gerosa, 2017).

With regard to this, there seems to be a consensus that the teaching of SE must go beyond the format of traditional lectures and encompass other teaching and learning approaches. Some authors

Figure 1. Development of competencies in the model (Portela, 2018)
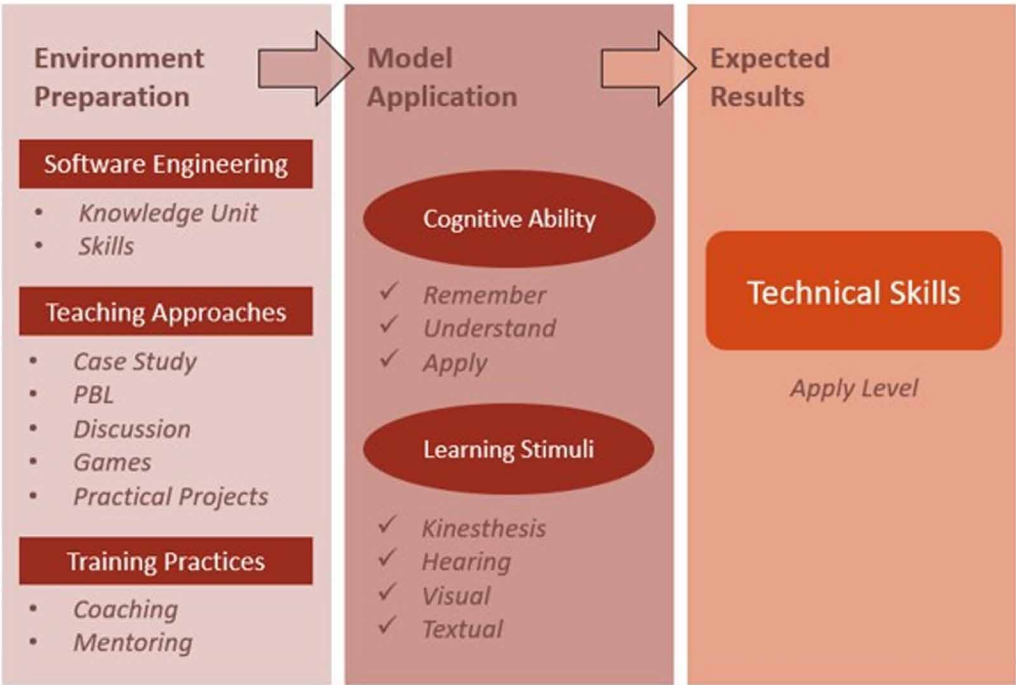
believe that practical teaching strategies are the most suitable methods for SE (Prikladnicki *et al.*, 2009)(Malik and Zafar, 2012)(Marques, Quispe and Ochoa, 2014)(Santos *et al.*, 2014). These authors also emphasize that this teaching should be more student-centred so that it can increase the students' motivation, strengthen their participation, and improve their relationship with other students and, hence their learning.

The use of games and the adoption of industry practices allow students to be motivated, increase their participation and interaction with the class, which has a positive impact on their learning and, consequently, development of their skills.

However, it is far from trivial to bring software projects developed inside a software company to the context of a classroom due to license issues or the possibility to disclose sensitive information (Pinto, Steinmacher and Gerosa, 2018).

Portela (2018) defines 7 (seven) principles that guide the practical teaching of SE, namely: (1) The students should be the focal point of the learning process; (2) Students should have different learning styles; (3) Learning should be based on problem solving; (4) Undertaking practical projects allows students to apply their knowledge; (5) The teaching approach must be iterative; (6) Training in industry should focus on the development of skills; and (7) Skills should be differentiated in levels of difficulty.

On the basis of these considerations, Portela *et al.* (2018)(2017) designed a model that seeks to maximize the benefits of adopting practical approaches that are combined with SE education. This model establishes a set of well-defined stages, which are arranged in an iterative cycle so that they can cover different learning profiles. This cycle is based on the experiential learning theory of Kolb (1984) and on the iterative teaching methodology proposed by Gary *et al.* (2013). The profiles are based on the classification of learning styles carried out by Fleming and Mills (1992). Figure 1 shows how skills are developed in the model designed by Portela (2018).

In this model, the students are at the centre of the teaching-learning process. They should adopt a participatory approach, have an active voice in decision-making and seek to solve the problems set by the teacher in a creative and original way by applying the knowledge they have acquired when carrying out practical activities.

The teacher must design learning situations and activities that can stimulate and challenge their students. He/she should act as a facilitator in the teaching-learning process.

The techniques, methods and teaching resources selected for this SPC teaching approach were based on the work of Portela *et al.* (2017). This model focuses on the following: the use of technical articles with reports of experiences and videos with essential features, and a combination of problem-centred learning (PBL), discussions of practical cases, the use of games, simulators or dynamics, undertaking practical projects and reflection. It is hoped that this approach will act as a stimulus to students and engage them more fully in a wide range of learning styles. As a result of this focus, the use of practical strategies and industrial training practices (i.e. the teacher must act as part of an Industry Mentoring Program), should enable students to actively participate in the teaching-learning process and to more efficiently develop the expected technical skills.

Each of the 4 units of the subject consists of 6 stages:

1. **Initiation:** The first lesson of each unit begins by outlining a problem. For example: "Is it possible for all the products to conform to a standard? If so, what about the product variations?";
2. **Preparation:** This stage runs parallel with all the other stages and is carried out in an extramural way. It involves the students studying the material provided by the teacher (videos, articles and books) as a means of understanding the topics. Thus, it can be driven by both textual and visual stimuli;
3. *Discussion***:** This stage consists of a traditional class taught by the teacher, followed by a discussion of the subject and is designed to help the students find ways to overcome their difficulties in carrying out practical tasks;
4. *Practice***:** Students practice the knowledge they have acquired by playing games. The objective of this stage is to allow the students to internalize and develop the necessary skills, as well as to benefit from their interaction and communication with the other students;
5. **Contextualization**: After completing the previous stages, the students will now finally undertake a practical project so that they can integrate all the skills they have acquired in the unit. In addition to the technical skills, this experience allows them to obtain client negotiation skills as well as to be involved in group work, improve their communication and tackle problems;
6. **Reflection:** The final stage involves students submitting the results obtained in the practical project and reflecting on their experience, by answering four questions based on the Scrum Sprint Retrospective ceremonies (What methods and techniques were applied when carrying out the project? What were the main difficulties experienced by the team? What methods and techniques were not applied by the team but could have helped? What changes should the team make when they re-run the project?).

The way these stages will be reflected in the teaching strategy of each unit is defined in accordance with the level of learning required for the topic, where: topics with the expected level of "*Remember*" will be addressed throughout the *Discussion* stage; topics listed as "*Understand*" require the Practice stage to be accomplished; and topics where the student is expected to reach the "*Apply*" level, will be covered in the Contextualization stage. It should be stressed that each unit undergoes all the stages of the cycle.

RELATED WORKS

The work of Bober and Zgodavová (2011) aims to outline a software system for the teaching of SPC. This involved designing, an interactive simulator for the control charts.

The authors were led to do this when they noted that their undergraduate students had little practical knowledge of the contents of SPC. Thus, the reason for designing the simulator was to overcome this shortcoming by carrying out practical activities and allowing students to acquire these new skills through hands-on experiences.

The main purpose of the simulator was to allow students to solve common problems encountered when working with the SPC process.

Among the range of simulators and software systems, there is also the work of Konrath *et al.* (2013), which states that one of the main techniques of the SPC are the control charts, since they provide essential knowledge to engineers and professional technicians.

However, the authors point out that the licenses of nearly all the software that is simple and easy to use are extremely expensive, which means they tend to be inaccessible to freelance professionals, small businesses and especially students. Thus, it was decided in this work to use GNU R, which is free open-source software and, in this way, provide a means of teaching the use of control charts.

The study by Johnson (2011), adopts a more light-hearted approach to the teaching of control charts through the use of paper helicopters. The point of this is to enable students to experience an unstable process. This instability is caused by imprecision or an incomplete product design and the student must find out how the process can be stabilized.

It also allows the students a) to understand the common causes of variability and thus be able to differentiate them from the particular causes of variability that result from incomplete design specifications and b) discover how standardization reduces process variability.

Other attempts to adopt more playful approaches can be found in the work of Jones, Hawkins and Smith (2008), where a data game is devised with the aim of teaching the use of control charts.

This paper outlines a simple method for the classroom, with the aim of demonstrating what process improvement looks like when displayed in an X-Bar R chart, which is a type of control chart. The method uses a dice game to bring about improvements in a manufacturing process. The students are taught how to construct and interpret control charts with an emphasis on how to link the results of the data to the process activities they routinely encounter in their workplaces.

The study of these related works made it possible to observe the combined use of traditional classes with other instruments in the strategies employed for simulation-based experimental learning.

Finally, all the studies examined here, are based on a single teaching approach, whether they involve games, simulators or practical projects. Also is notable the low number of studies on the SPC teaching.

EVALUATION

The objective of the study was to assess the efficacy of SPC learning at the application level and how it results from the new teaching approach, in contrast with what is achieved by traditional classes in undergraduate Computing program

Research and Evaluation Strategy

A formal experiment was conducted (a pre-test/post-test randomized control group design), with the aim of comparing and evaluating the effectiveness of different types of learning activities (Cohen *et al.*, 2000). This experimental design allowed a statistical comparison to be made between the behaviour observed in the experimental group with that observed in the control group (Campbell and Stanley, 1963). The experiment was carried out as follows:

1. At the beginning of the experiment, the students did a pre-test that covered the contents of SPC and answered a questionnaire about their personal background and degree of motivation. This information served as a means of ensuring that the random division of participants into control group and experimental group, had achieved an acceptable balance. Thus, for statistical purposes, the two groups could be considered to be equivalent;
2. The interventions (the new teaching approach and the traditional classes) were made. The experimental group was shown the learning activities planned for the new teaching approach, while the control group attended traditional classes[1];
3. After the conclusion of each unit studied, both groups answered questions in a post-test about the contents learned.
4. At the end of the experiment, the students, from experimental group, completed a perception questionnaire about the learning experience.

Table 3 summarizes this information about the design of the experiment.

Table 3. Summary of the experiment

Groups	Preparation	Interventions		Conclusion
Control	Pre-test application; Background and motivation questionnaire; Allocation of groups.	Traditional classes.	Post-test application	Questionnaire on experience
Experimental		Classes based on planned learning activities and support material.		

RESEARCH QUESTIONS AND HYPOTHESES

As a means of assessing the level reached by the students, Table 4 displays the scores obtained, based on the criteria for correctness applied to the students. Again, it should be pointed out that the word Apply includes *Understand*, which in turn includes *Remember* (Nunes *et al.*, 2016).

It should be noted that achieving a score of 3 means that the student was able to remember, understand and apply the knowledge acquired in that unit. Students who obtained Score 2 are able to understand and Score 1 is for students who have just reached the *remember* level.

Table 4. Scoring Rubric

Score	Criterias
3	The student tackled the problem correctly using the technique that had been taught
2	The student was able to justify the technique chosen to solve the problem
1	The student was able to cite the technique required to solve the problem
0	The student did not answer the question

Table 5 shows the objective of the study, its research questions and instruments, together with the null hypotheses. The Y.1 to Y.4 variables are directly related to the four units taught and were evaluated to determine the cognitive levels of learning (Application), reached by the students.

Table 5. Detailing of the study objective 1

Study objective
Research question 1: What is the efficacy of Statistical Process Control learning when the proposed approach is adopted at the application level?
Hypothesis H01: There will be no difference between the pre and post-test scores obtained by the Experimental group (the members of the group will have the same skills) at the application level

Variables	
Y.1. Software Process	Y.1. Software Process
Y.2. Measurement	Y.2. Measurement
Y.3. Statistical Control	Y.3. Statistical Control
Y.4. Process Improvement	Y.4. Process Improvement

Formulation: post(Mpos; A) > pre(Mpre; A), where:

1. A = Experimental Group
2. Each student's pre and post-test scores are:

$$Npre_i = \frac{100\left(Y1 + Y2 + Y3 + Y4\right)}{4} \text{, where } i \text{ is a Group A student}$$

$$Npos_i = \frac{100\left(Y1 + Y2 + Y3 + Y4\right)}{4} \text{, where } i \text{ is a Group A student}$$

3. Average student scores by group:

$$Mpre = \frac{\sum_{i=1}^{m} + Nprei}{m} \text{, where } m \text{ is the number of students in Group A}$$

$$Mpos = \frac{\sum_{i=1}^{m} + Nposi}{m} \text{, where } m \text{ is the number of students in Group A}$$

Instruments: Pre and post tests
Research question 2: What is the efficacy of Statistical Process Control learning when the new approach is adopted when compared with that of traditional classes?
Hypothesis H02a: There will be no difference in the pre-test scores between the Control group and the Experimental group with regard to the level of cognition reached (Application).

Variables
Y.1, Y.2, Y.3 e Y.4 with their respective scores, as described in Research Question 1

Formulation: pre(Mpre; A) > pre(Mpre; B), where:

1. A or B = Experimental or Control Group
2. Each student's pre-test scores are:

$$Npre_i = \frac{100\left(Y1 + Y2 + Y3 + Y4\right)}{4} \text{, where } i \text{ is a Group A or B student}$$

3. Average student scores by group:

$$Mpre = \frac{\sum_{i=1}^{m} + Nprei}{m} \text{, where } m \text{ is the number of students in Group A or B}$$

Table 5. Continued

Instruments: Pre-test
Hypothesis H02b: There will be no difference in the post-test scores between the Control group and the Experimental group with regard to the level of cognition reached (Application).
Variables
Y.1, Y.2, Y.3 e Y.4 with their respective scores, as described in Research Question 1

Formulation: post(Mpost; A) > post(Mpost; B), where:
  1. A or B = Experimental or Control Group
  2. Each student's pre-test scores are:
 	
$$Npos_i = \frac{100\left(Y1 + Y2 + Y3 + Y4\right)}{4}$$, where i is a Group A or B student
 	
  3. Average student scores by group:
 	
$$Mpos = \frac{\sum_{i=1}^{m} + Nposi}{m}$$, where m is the number of students in Group A or B

Instruments: Post test

Instrumentation

The pre and post anonymous tests were used to collect the data needed to answer Research Questions 1 and 2. At the application level, the test consisted of practical problems that had to be solved. For this reason, the activities were contextualized in terms of the real conditions of a software organization, and in accordance with the content taught in each Unit. They provided opportunities for following the necessary stage to statistically control a process.

The same question were applied for the pre and post-tests and were corrected by two specialists in the area who were not involved in content teaching, with the help of a third one who could arbitrate in case of disagreements. The scores of this test were calculated according to variables Y.1 to Y.4, as described in Table 5.

The participants in any of the tests gave no information. The scores of all these tests were only made available to the students at the end of the experiment.

Disturbance factors were noted in an objective questionnaire about student motivation (e.g. the Importance of SPC and Interest in learning more about the subject).

Execution

The experiment was carried out in the first semester of 2018, in an optative subject offered to the academic community doing the Computer Science program at Federal University of Amapá, Brazil.

All the participants of the experiment were volunteers and the class had 40 students enrolled, (20 students in each group). The course lasted for 7 days and each group held 4 in-person classes, each lasting up to 4 hours. Between each day of the classes, there was a day off for students to study the provided materials. The schedule for the experiment is described in Table 6. During the enrolment procedure, the students completed a motivation questionnaire. After the experimental and control groups interventions and their respective post-tests, the students answered the feedback perception questionnaires.

During the execution of the interventions, each group had its own specific teacher. Thus, only one teacher gave a class to the control group, while another teacher taught the experimental group. This decision was made to minimize the threat of the teacher teach a more dedicated class for the

Table 6. Experiment schedule

Day	Control Group	Experimental Group
During the on-line registration	Pre-test; Questionnaire on the motivation and background of the applicants.	Pre-test. Questionnaire on the motivation and background of the applicants. Students in the Experimental group supplied with the support materials for Unit 1.
1	Traditional lecture on Unit 1; Post-test.	Class with the new teaching approach adopted for Unit 1; Post-test.
2		Study of the support materials needed for the next unit
3	Traditional lecture on Unit 2; Post-test.	Class with the teaching approach proposed for Unit 2; Post-test.
4		Study of the support materials needed for the next unit
5	Traditional lecture on Unit 3; Post-test.	Class with the teaching approach proposed for Unit 3; Post-test.
6		Study of the support materials for the next unit.
7	Traditional lecture on Unit 4; Post-test.	Class with the teaching approach proposed for Unit 4; Post-test.
On-line, after the last day	Perception Questionnaires on the suitability of the approach.	Perception Questionnaires on the suitability of the approach; Feedback questionnaire.

experimental group compared to the control group. It is also be emphasized that these two teachers worked together in preparing the lessons and designing a lesson plan.

DATA ANALYSIS

This section analyses the data obtained by the execution of the experiment detailed in the previous sections. This analysis will be performed from the research questions.

Analysis of Research Question 1

The Shapiro-Wilk normality test (1965) was conducted to determine if the scores obtained by the students in the evaluation had a normal distribution. With regard to the experimental dataset, the normality test application resulted in a W equal to 0.91 in the Experimental Group and equal to 0.96 in the Control Group, with the dataset being normally distributed for $p < 0.05$.

Given the normality of the data and the fact that it was our objective to measure the difference between pre-test and post-test in the same population, we chose the Student two-tailed t-test for paired samples in hypothesis H01.

For H01 "There will be no difference in the pre and post-test scores in the averages obtained by the Experimental group (since the group will have the same skills) at the application level", the Experimental group scored 0.04 ± 0.08 in the pre-test, while the same group scored 2.50 ± 0.37 in the post-test. Thus, the mean gain of $\Delta = 2.46$ in the Experimental group indicates there was

an incremental learning of this group, since the p-value of 0.01 is statistically significant. Table 7 summarizes the results obtained.

Analysis of Research Question 2

In the case of RQ2, "What is the learning effectiveness of SPC through the developed approach compared with that of traditional classes?" In the pre-test, a comparison was made between the Experimental Group (0.04 ± 0.08) and the Control Group (0.06 ± 0.13). Thus, the stated hypothesis (H02a), "There will be no difference in the pre-test scores between the Control group and the Experimental group with regard to the level of cognition reached", was not refuted.

From this result, the null hypothesis H01 was rejected. Figure 2A shows the comparison between the mean averages obtained by the Experimental group for the cognitive application level, with a detailed account of each study unit. For purposes of comparison, Figure 2B shows the same data for the Control group.

Table 7. Efficacy of learning based on the scores obtained in the practical project

Variables	Pre-test	Post-test
Sample size	20	20
Minimum	0.00	1.75
Maximum	0.25	3.00
Medium	0.00	2.50
First quartile	0.00	2.00
Third quartile	0.00	2.75
Average	0.04	2.50
Standard deviation	0.08	0.3794

Figure 2. (a) Averages obtained per Unit for the Experimental Group at the cognitive application level; (b) Averages obtained per Unit for the Control Group at the cognitive application level

In the post-test stage, the student two-tailed t-test was chosen to make a comparison between the Experimental and Control Group, because of its independent samples. This took account of the normality of the data and its objective was to evaluate the difference between two populations with treatment conditions and two samples (i.e. of treatments). The Experimental group scored 2.50 ± 0.37 in the post-test, while the Control group scored 1.38 ± 0.62 in the post-test. This means that there is a real difference between the groups, where the Experimental score is $\Delta = 1.12$ higher than

Control and suggests there was an increase in the learning of this group, since the p-value of 0.01 is statistically significant.

On the basis of this result, it was possible to reject the H02b hypothesis; "There will be no difference in the post-test scores between the Control group and the Experimental group with regard to the level of cognition reached (Application)". Table 8 summarizes the results obtained by the Experimental group and those obtained by the Control group for the cognitive application level and provide a detailed account for each study unit.

Table 8. Comparison of learning effectiveness among the participating groups

Variables	Experimental Group		Control Group	
	Pre-test	Post-test	Pre-test	Post-test
Sample size	20		20	
Minimum	0.00	1.75	0.00	0.25
Maximum	0.25	3.00	0.50	2.50
Sum of points	3	195	5	111
Medium	0.00	2.50	0.00	1.50
First quartile	0.00	2.00	0.00	1.00
Third quartile	0.00	2.75	0.00	1.75
Average	0.04	2.43	0.06	1.38
Standard deviation	0.08	0.37	0.13	0.62

RESULTS DISCUSSION

The results obtained from testing the H01 hypothesis, "There will be no difference in the averages of the pre and post-test scores obtained by the Experimental group (i.e. the group will have the same skills) at the application level", suggests that the planned approach for the teaching of SPC has a positive effect on learning effectiveness at the cognitive application level.

In addition, the rejection of the H02a and H02b hypotheses suggests that this approach has a learning effectiveness greater than the efficacy obtained by the traditional expository classes, since although the averages of pre-test gains were not statistically significant, the average gains obtained by the Experimental group in the post-test were significantly higher than those of the Control group.

These results in H01 and H02b can be attributed to the fact that the teaching strategies applied to the experimental group were sharply focused on the students, and carrying out practical activities, as well as being based on problem-solving. In view of this, the results obtained are in line with what several authors have pointed out, which is that practical strategies seem to be the most suitable for the teaching of SE and are preferred by the students (Prikladnicki *et al.*, 2009)(Malik and Zafar, 2012) (Marques *et al.*, 2014)(Santos *et al.*, 2014) and similar to experiences using coaches (Rodrigues, Soria, and Campo, 2016) or Scrum masters to support student teams (Scharf and Koch, 2013),

However, experience dictates that students with higher grades are, in general, more willing to undertake extra activities than those having lower grades (Gamo, 2018). In this study, no analysis was made on the grade history of the participating students. Thus, it is not possible to totally discard this possibility.

There was also a statistically significant increase in the average score of the Control Group between the pre and post-test, with a gain of $\Delta = 1.32$. This suggests that, although the proposed approach has achieved better results, both approaches can have positive effects on student learning.

This statement can be further corroborated by the lack of data to refute the H02a hypothesis, "There will be no difference in the pre-test scores between the Control group and the Experimental group with regard to the level of cognition reached". This is because the pre-test score of both groups tended to be zero and provides evidence that everything the students learned about SPC was acquired through one of the approaches.

With regard to the pre-test scores, this fact can perhaps be attributed to the very nature of the SPC, since it is a complex topic and little known even by post-graduate software engineers (Alhassan and Jawawi, 2014). However, these initial scores may show that even when the groups are randomly divided, they are statistically equivalent.

When considering the student motivation to learn SP, it was observed that most students believe that learning the topic is important. However, some students did not realize the importance of this advanced SE topic. This problem may explain some of the low scores obtained by the control group. So, in this case, the teaching approach itself may not be responsible for these results.

Also, the Control group had a large workload of theoretical classes. It is possible that the students view disconnects between homework assignments and the project, and between the projects and lecture. Resulting in a lower score than those who were in the Experimental group.

During the experiment, it was possible to observe some weaknesses in the approach. One of them was the fact that the teaching approach is extremely student-centred. This meant that very few lectures were. However, this is an expected feature of this teaching approach when applied to the experimental group, as it is expected that the knowledge required for the practical activities will be acquired through the study of the support material provided by the teacher (outside the classroom). The teacher will act as a mentor during the learning activities that are designed to internalize knowledge and thus enable the students to reach the application level for the concepts learned. Further studies will be developed to better determine the learning preference profiles that best fit the proposed teaching approach.

Another weakness observed concerns the limited time available to undertake the activities. Thus, a review of the learning activities could reduce the amount of work required from the students and make it possible to carry out any activity within the scheduled time frame. The intention is for the homework assignments to introduce students to the technologies and teamwork, but this can gets lost in the aggressive deadlines. It is worth mentioning that the 7-day course completion period was adopted to facilitate the execution of the experiment. The teacher who chooses to teach the subject based on the developed approach, may choose to dilute the classes and activities over a semester. Which can lessen this weakness.

Also, some students cited as a weakness the difficulty of understanding more complex subjects. As stated earlier, this may be a limitation of the learning preferences of these students who do not get along with self-taught approaches. As it may also be a failure of the teaching approach. One solution for this would be to include new and better video lessons and articles, as a means of overcoming this difficulty. Again, further studies are needed to deepen the problem and its possible solution.

Despite the good results, there were dysfunctional teams who did not work well together, and personal conflicts may have jeopardized the scores.

THREATS TO VALIDITY

This section presents and discusses some threats to the validity of the study.

Internal Validity

In the case of the experiment, some possible threats to internal validity may be owing to the fact that the groups behave differently as a result of a diversity of background and motivation, or even be an effect of the students´ own instrumentation.

In an attempt to reduce the influence of disturbance factors, the groups were randomly divided by drawing lots and this served to make the control group and the experimental group statistically equivalent.

As for the question of maturity, this problem may have occurred because the learning activities had not been planned for the experiment in advance, or from the student's own initiative in seeking more information about the content. As a means of mitigating the influence of this factor, the participants were instructed not to study or work on issues that were not among the activities planned for the experiment. The members of the Control group were also instructed not to consult the support material that was supplied to the experimental group. However, there is no way to monitor this behaviour and ensure that the students actually followed these instructions.

The internal risk of ¨instrumentation¨ could occur if the same authors who applied the test were those who conducted the analysis of the data. However, the authors were careful to define in advance who would carry out the data collection and analysis and ensure that this person would only have a single responsibility.

Hence an attempt was made to reduce the influence of subjective factors during the evaluation of the practical tasks, and all the tests that were done in pairs underwent a correction procedure that was supported by feedback. When making this correction, the two evaluators were not given any information about the participants. Thus, although the average gains were statistically significant for H01 and H02b, it is possible that these results had been achieved not only by the merit of the treatment application and may in part have involved a regression to the mean (Campbell and Stanley, 1963).

Again, with regard to instrumentation, since the pre and post-tests comprised identical questions, there is a risk that the learning could have been achieved through the repetition of the test itself. However, owing to the low scores achieved in the pre-test and the fact that the participants were only awarded their grades at the end of the experiment, this risk may have been lessened during the experiment.

There is also the fact that each of the two teachers has taught the subjects in their respective groups. This experimental design and decision may have led one group to learn less than the other not owing to the effects of the approach itself but rather because of the depth of knowledge of the group teacher. As a way of overcoming this problem, the teachers worked together throughout the planning phase.

External Validity

These results can only be generalized and restricted to the academic world. There is also the fact that the experiment was carried out with a small sample of participants and, so far, has not been replicated in groups of students in other universities. These factors make it even more difficult to generalize the results obtained. Moreover, the group used for the evaluation did not include a trained Software Engineer, and thus may not reflect the positive results that are obtained by another target audience. However, having a small sample was a necessary decision to make sure the experiment could be carried out.

The purpose of the teaching approach that was adopted for the experimental group, was to encourage the students to become involved in the practical application of the concepts learned, but despite this objective, the activities were based on simplified real-world scenarios. Hence, there is no guarantee that the acquired skills will actually be reflected in a real software development process.

Construction Validity

The potential effects of learning were measured by comparing the results obtained by the control group with those achieved by the experimental group. However, this kind of methodology may not be enough to measure the effects of real learning in terms of the level of application. Thus, it cannot be claimed that the participants learned how to use this newly acquired knowledge in a setting that was different from that used during the experiment.

Conclusion Validity

Simpson's Paradox is a phenomenon in which a trend appears in different data groups and disappears when these data are gathered into a single group and thus a new trend emerges from this new aggregate dataset. This is a potential danger that may have occurred in this experiment.

Despite this, even if all the data collected by the two groups participating in this experiment were collected, the sample would remain very small. In view of this, no valid statistical relationship can be demonstrated. To circumvent this obstacle, a more robust statistical test was conducted, where a low statistical power was accepted. This strategy has already been adopted in other similar studies (Chaves *et al.*, 2015) (Wangenheim *et al.*, 2009).

CONCLUSION

The results of this study provided preliminary evidence that the approach to teaching SPC put forward here, achieves a greater degree of learning effectiveness at the application level than classes based on the kinds of instructional methods employed in traditional lectures.

This work can be regarded as an explanatory study that is aimed at obtaining an insight into the learning effectiveness of the approach examined here, as well as its drawbacks, and making suggestions for how it can be improved by the participants. The results were significant, since the participants achieved positive results in SPC learning and it was possible to obtain feedback with regard to the strengths and weaknesses of the approach and make recommendations for improvements.

Also, the positive results obtained in this experiment are significant because they are consistent with the learning gains obtained in previously performed experiments, as can be seen in Furtado and Oliveira (2018b)(2019).

However, owing to the threats to validity that were found in the experiment, the results cannot be generalized to all application contexts. Finally, although these results were significant, further studies are still needed to investigate the different outcomes that may occur when other student populations adopt the approach.

REFERENCES

ACM/IEEE. (2013). *Computer science curricula 2013. Curriculum guidelines for undergraduate degree programs in Computer Science*. Author.

Alhassan, M. A., & Jawawi, D. N. (2014). Sequential Strategy for Software Process Measurement that Uses Statistical Process Control. *8th Malaysian Software Engineering Conference (MySEC),* 37-42. doi:10.1109/MySec.2014.6985986

Bloom. (1956). *Taxonomy of Educational Objectives: The Classification of Educational Goals. Handbook I, Cognitive Domain*. Longmans.

Bober, P., & Zgodavová, K. (2011). Education web tool for Statistica Process Control. *14th International Conference on Interactive Collaborative Learning (ICL2011) 11th International Conference Virtual University (vu'11)*.

Campbell, D. T., & Stanley, J. C. (1963). *Experimental and quasi-experimental designs for research*. Houghton Mifflin Company.

Chaves, R. O., Wangenheim, C. G. V., Furtado, J. C., Oliveira, S. R. B., Santos, A., & Favero, E. L. (2015). *Experimental Evaluation of a Serious Game for Teaching Software Process Modeling.* IEEE Transactions on Education. doi:10.1109/TE.2015.2411573

Cohen, L., Manion, L., & Morrison, K. (2000). *Research methods in education*. Routledge Falmer.

Fleming, N. D., & Mills, C. (1992). Not Another Inventory, Rather a Catalyst for Reflection. *To Improve the Academy, 11*, 137.

Furtado, J., & Oliveira, S. (2017). A study on perception of the usefulness of statistical process control in software development organizations: an application of survey. *14th CONTECSI - International Conference on Information Systems and Technology Management*.

Furtado, J., & Oliveira, S. (2018a). A Methodology to Teaching Statistical Process Control in Computer Courses. *13th International Conference on Evaluation of Novel Approaches to Software Engineering*. doi:10.5220/0006800504240431

Furtado, J. C., & Oliveira, S. R. B. (2018b). Evaluating Students' Perception of their Learning in a Student-Centered Software Engineering Course – A Experimental Study. *13th International Conference on Software Technologies*.

Furtado, J. C., & Oliveira, S. R. B. (2019). An Experimental Evaluation of a Teaching Approach for Statistical Process Control for Software Engineers – A Experimental Study. *14th International Conference on Software Technologies*. doi:10.5220/0007772501530161

Gamo, J. (2018). Assessing a Virtual Laboratory in Optics As a Complement to On-Site Teaching. *IEEE Transactions on Education*, *62*(2).

Gary, K., Lindquist, T., Bansal, S., & Ghazarian, A. (2013). A Project Spine for Software Engineering Curricular Design. *Proceedings of 26th Conference on Software Engineering Education and Training (CSEET),* 299-303. doi:10.1109/CSEET.2013.6595265

Johnson, D. (2011). Using Paper Helicopters to Teach Statistical Process Control. *Decision Sciences Journal of Innovative Education*, *9*(9).

Jones, M. P., Hawking, R. S., & Smith, R. (2008, Spring). Enhancing student understanding of Control Charts using a dice activity. *The Technology Interface*.

Kolb, D. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice-Hall.

Konrath, A. (2013). Applications in Teaching Statistical Quality Control with different R interfaces. *2013 IEEE Global Engineering Education Conference (EDUCON).* doi:10.1109/EduCon.2013.6530099

Malik, B., & Zafar, S. (2012). A Systematic Mapping Study on Software Engineering Education. International Journal of Social, Behavioral, Educational, Economic. *Business and Industrial Engineering*, *6*(11), 3343–3353.

Marques, M. R., Quispe, A., & Ochoa, S. F. (2014). A Systematic Mapping Study on Practical Approaches to Teaching Software Engineering. In *Frontiers in Education Conference*. Madrid: IEEE. doi:10.1109/FIE.2014.7044277

Nunes, D. J., Yamaguti, M. H., & Nunes, I. (2016). Refinement of student competences of the software engineering course. *IX Forum on Education in Software Engineering*, 143-146.

Paranthaman, D. (1990). *Controle de qualidade*. McGraw-Hill Ltda.

Pinto, G., Steinmacher, I., & Gerosa, M. (2017). Training software engineers using open-source software: The professors' perspective. *30th Conference on Software Engineering Education and Training (CSEET),* 117–121. doi:10.1109/CSEET.2017.27

Pinto, G., Steinmacher, I., & Gerosa, M. (2018). Leaving behind the software history when transitioning to open source: Reasons and implications. *Proceedings of the 14th International Conference on Open Source Systems*. doi:10.1007/978-3-319-92375-8_5

Pinton, D. H. (1997). *Controle estatístico de processo*. São Paulo: Rev IMES.

Portela, C. (2018). *Um Modelo Iterativo para o Ensino de Engenharia de Software Baseado em Abordagens Focadas no Aluno e Práticas de Capacitação da Indústria*. Tese de Doutorado, Universidade Federal de Pernambuco.

Portela, C. S., Vasconcelos, A. M. L., Oliveira, S. R. B., & Souza, M. (2017). The Use of Industry Training Strategies in a Software Engineering Course: An Experience Report. *The 30th IEEE Conference on Software Engineering Education and Training*.

Prikladnicki, R., Albuquerque, A., Wangenheim, C., & Cabral, R. (2009). Teaching software engineering: challenges, teaching strategies and lessons learned in Proceedings of II Forum on Education in Software Engineering, Fortaleza, Brazil.

Rocha, A., Souza, G. and Barcellos, M. (2012). Medição de Software e Controle Estatístico de Processos, Ministério da Ciência, Tecnologia e Inovação, 232 p, Brasília – BR.

Rodrigues, G., Soria, A., & Campo, M. (2016, August). Measuring the impact of agile coaching on students' performance. *IEEE Transactions on Education*, *59*(3), 202–209. doi:10.1109/TE.2015.2506624

Santos, R. (2014). Ferramentas, Métodos e Experiências no Ensino de Engenharia de Software: um Mapeamento Sistemático. *Anais do III Congresso Brasileiro de Informática na Educação*, 544-548.

Santos, S., Alexandre, G., & Rodrigues, A. (2015). Applying PBL in Project Management Education: a Case Study of an Undergraduate Course. *Proceedings of Frontiers in Education Conference (FIE)*, 1-8. doi:10.1109/FIE.2015.7344232

Scharf, A., & Koch, A. (2013). Scrum in a software engineering course: An in-depth praxis report. *Proc. 26th Int. Conf. Softw. Eng. Educ. Train.*, 159–168.

Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika, London*, *52*(3-4), 591–609. doi:10.1093/biomet/52.3-4.591

SOFTEX. (2016). *MPS.BR – Guia Geral MPS de Software:2016*. Disponível em: www.softex.br

Software Engineering Institute. (2010). *CMMI® for Development, Version 1.3, Improving processes for developing better products and services, No.CMU/SEI-2010-TR-033*. Carnegie Mellon University.

Wagner, C. H. (1982). Simpson's Paradox in Real Life. *The American Statistician*, *36*(1), 46–48.

Wangenheim, C. G., Thiry, M., & Kochanski, D. (2009). Empirical evaluation of an educational game on software measurement. *Empirical Software Engineering, Kluwer Academic Publishers Hingham, MA, USA*, *14*(4), 418–452. doi:10.1007/s10664-008-9092-6

ENDNOTE

[1] For this study, traditional classes are considered to be those based on content exposure by the teacher, supported by any activities/content fixing exercises.