Received: 17 April 2023 | Revised: 29 August 2023 | Accepted: 5 September 2023

DOI: 10.1002/cae.22681

RESEARCH ARTICLE

WILEY

Automated and continuous assessment implementation in a programming course

Kevin Calderon | **Nicolás Serrano** | **Carmen Blanco** | **Iñigo Gutierrez**

TECNUN School of Engineering, University of Navarra, Pamplona, Spain

Correspondence
Kevin Calderon, Paseo de Manuel Lardizabal, N° 13, 20018, Donostia, San Sebastián, Spain.
Email: kcalderon@unav.es

Abstract

Continuous assessment is an assessment methodology whose objective is to assess students on an ongoing basis. However, designing, organizing, correcting, and evaluating continuous assessment increases the workload of teachers. Moreover, this methodology may not promote deep learning if it is not implemented properly. In this study, we implemented continuous assessment in an undergraduate programming subject using an automated assessment tool to reduce the workload of professors. We used design-based research (DBR) to implement a prototype of assessment methodology which includes an automated assessment tool developed by our research group. DBR provides us with a scientific background for this implementation through an iterative process in which we progressively come to assess all the activities that students perform in the course. In the different iterations of this process, we have collected students' final and project grades, and their opinions through surveys about the assessments we have implemented. These results allow us to demonstrate that the performance of at least two types of students improves after the implementation of continuous assessment, while at the same time, the depth of learning in the class is not affected. We have also found that students are more motivated and committed to the course when continuous assessment is used as they prefer automated assessment over the traditional exercises. In addition, the implementation of the continuous assessment has shown us some unexpected outcomes about flexibility in methodology design, collection of large amounts of data from the learning process, and students acquiring useful skills for programming. In reality, this can result in students gaining deeper knowledge if they are confronted with a greater number of situations during this time in which they test their knowledge.

KEYWORDS

automated assessment tool, continuous assessment, deep learning, design-based research, programming subject

1 | INTRODUCTION

In the education domain, the assessment methodology is the most influential element on the learner at the moment of choosing a study technique [2, 13]. Therefore, assessment is considered as a fundamental part of the learning process, that is, the different stages that the learner goes through to acquire new knowledge [4]. Continuous assessment (CA) is an assessment methodology, and it consists of constantly assessing students throughout the learning process. It encourages them to learn from the beginning of the course so that their commitment to learning is greater [8]. In fact, the literature also indicates that this methodology improves the grades of the students [6, 18, 23].

CA is not free of concerns: some studies suggest that it promotes superficial learning because it does not encourage students to assimilate concepts in their entirety, nor to consider the possible relationships between them or their application in other domains [14, 30]. However, other studies show that students find the new concepts easier to understand with this methodology since they are related to previous content that they have already studied, so it promotes deep learning [24]. Both statements could be explained as some approaches to implement CA can lead to superficial learning. For example, a common approach when implementing this methodology is to add midterm exams. However, these assessments might not be sufficient. Students distribute their time unevenly across courses, often focusing on topics associated with assessment and nothing else. Although the average performance of the students could improve, most of the students still follow the traditional study patterns: they only study in the days before each exam, losing the chance offered by CA to experience deep learning [10].

At the same time, adding more assessments does not seem to be an adequate solution as they increase the workload of professors because they involve several tasks such as designing, organizing, correcting, and grading [25]. In addition, the time professors spend on these tasks often goes unrecognized, despite positive outcomes such as increased student motivation and learning [32].

Automated assessment emerges as a solution to obtain the benefits of CA while avoiding the increased workload for professors, as it spares them from repetitive tasks related to assessment [20]. In this sense, automated assessment can help to cope with the increment of the workload, a main obstacle to implement CA.

In our faculty, the School of Engineering of the University of Navarra, several professors have introduced midterm exams as an approach to CA. However, this strategy can be considered as an incomplete implementation as it reproduced the already mentioned problems: most of the students have not achieved deep learning due to their inconsistent study behavior, and the workload of the professors has increased.

For this reason, our research group MENTOR (Methodologies in Education and New Technologies Orientation and Research), has developed a tool that allows the automated evaluation of different engineering exercises. After the initial development of the tool was finished, we talked with different professors to find the most suitable subject in which to use the tool. Although we will describe the tool in future sections, it allows, among other things, the automatic grading of programming exercises and, therefore we chose a subject in this area.

The next goal was to achieve the implementation of the CA in this subject with the developed tool in using a scientifically rigorous methodology. Thus, we proceed to define the following research questions as the framework of this study:

RQ1. *Does a custom automated assessment tool enable the implementation of CA in a programming subject to improve student performance?*
RQ2. *Does a custom automated assessment tool enable the implementation of CA in a programming subject without encouraging superficial learning?*
RQ3. *What is the opinion of the students of a programming subject regarding the implementation of CA with a custom automated assessment tool?*

2 | AUTOMATED ASSESSMENT IN ENGINEERING

As we mentioned earlier, we assess students on an ongoing basis when we implement CA. However, increasing the number of assessments is limited by the amount of work required of professors. Our goal is to test whether we can overcome this obstacle by using automated assessment. Automated assessment consists of implementing a process that allows the automatic correction and grading of student activities. To do this, we need to use a tool or system that performs these tasks automatically.

Technology is a great help in providing various automated assessment tools. In early implementations of automated assessment tools, there was a tendency to avoid complex forms of assessment due to the difficulty of developing algorithms that make them possible. However, these assessments do not force students to apply the concepts they have learned and could also lead to superficial learning [7, 16]. Some examples of the exercises included in these assessments are: single-choice or multiple-choice questions, fill-in-the-blanks, word or sentence ordering, and so forth.

There are certain learning outcomes for which automated assessment is not appropriate [31]. In public speaking, for example, the components of eye contact and body language are critical aspects of the speeches being assessed and are very difficult, if not impossible, to measure through online assessment. Nor does it seem easy to automatically assess learning outcomes related to the acquisition of certain personal or interpersonal skills, such as critical thinking or teamwork, and so forth. However, automated assessment may be more feasible when it comes to assessing more technical competencies, such as in the case of engineering, where most of the solutions to the problems posed in this field are numerical results, algebraic expressions, or algorithms.

These tools could be suitable for a specific type of exercise, such as the one described by Sangwin [27], which is suitable for assessing mathematical exercises. Alternatively, they can be open to the assessment of different types of scientific exercises [15]. In the case of programming education, we found different assessment tools developed for programming competitions and later used in classrooms, as in the studies by Restrepo-Call et al. [26] and Ferreira et al. [9].

Considering the information gained from these experiences, we obtained some guidelines for the development of a custom automated assessment tool focused on engineering education [29]. The tool named as Codex is a web platform based in a Java stack that allows different types of assessments depending on the needs of each subject. The goal was to obtain an automated assessment tool that could be adapted to a variety of exercises common in the engineering domain. It includes the automated assessment of math, science, and programming exercises with a certain level of complexity, as well as the assessment of simpler exercises such as those mentioned above.

This tool was the one used to implement CA, as it would make it easier for professors to implement it in their subjects without increasing their workload. The version of the tool used for this study was hosted on a university server and it can be accessed by both professors and students with a username and password. The interface is adapted to the type of user. On the one hand, the interface of the students allows them to answer the proposed exercises and receive immediate feedback (Figure 1). On the other hand, the interface of the professors allows them to introduce, configure, and manage exercises (Figure 2).

Since we can customize the tool, we can integrate it into the design process of the CA. In this way, it is possible to modify the tool and add new features as the evaluation methodology evolves. A more detailed description of the tool features is described by Serrano et al. [28]. The configuration and assessment process of each exercise is very similar. First, the teacher enters the information for each exercise as shown in Figure 2, including the statement of how to evaluate the answer of the student. Then the professor groups the exercises to be assessed and configure the assessment for different scenarios: for study, for a practice, or for exams. Students access this group of exercises through an interface that allows them to enter and submit their answer, as shown

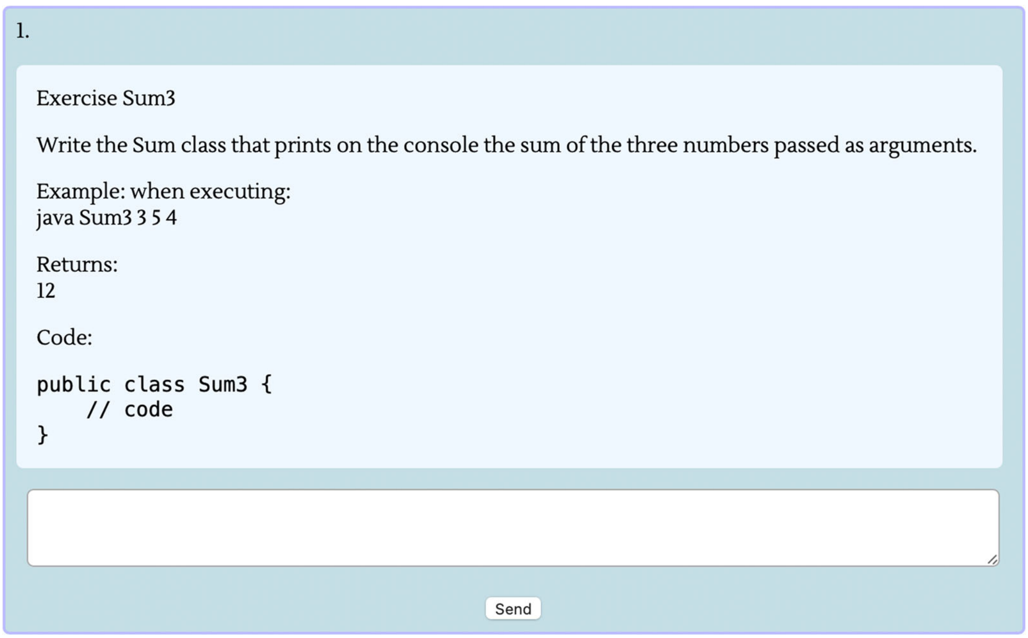


FIGURE 1 Exercise interface for the students.

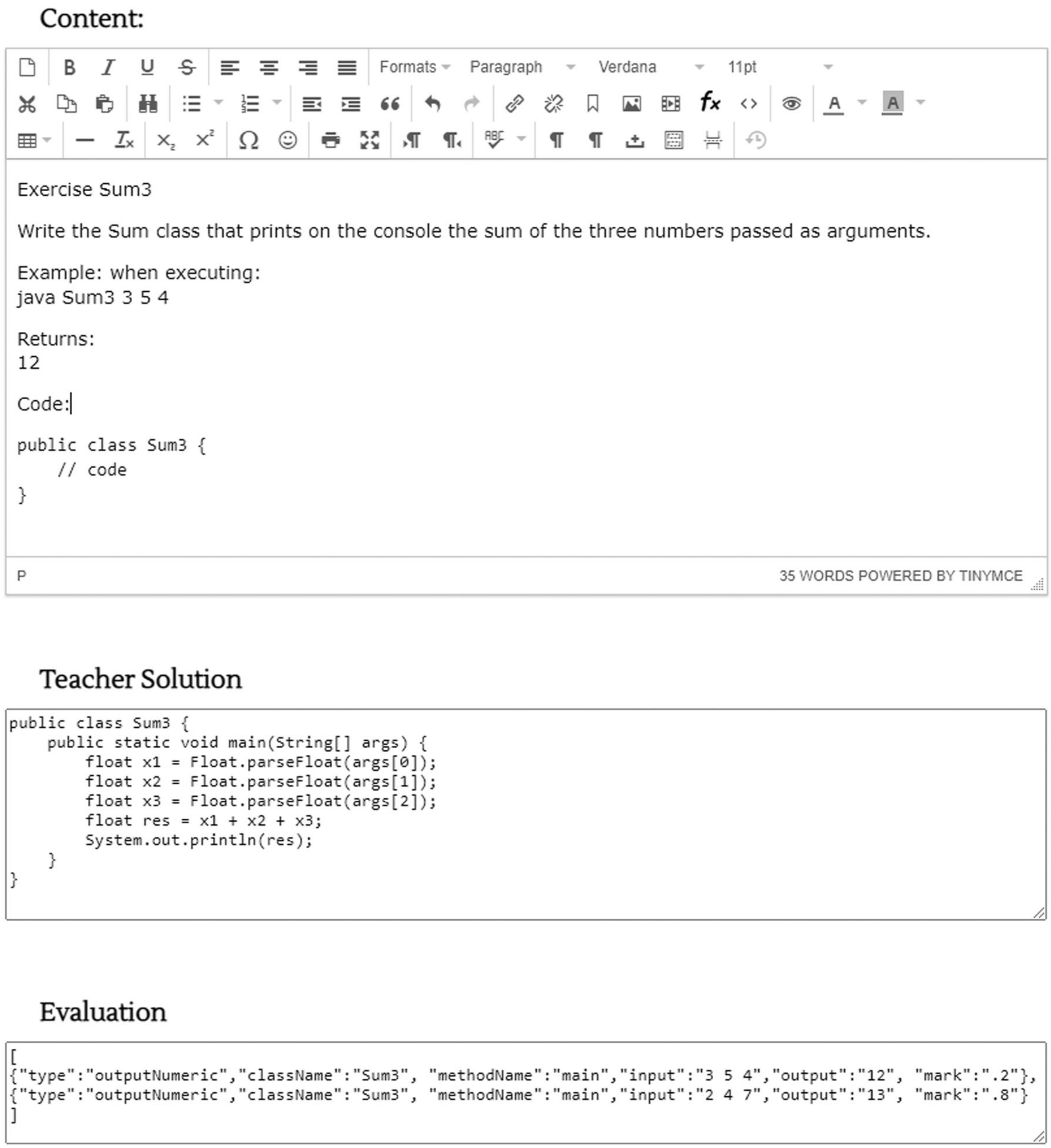


FIGURE 2 Exercise edition interface for professors.

in Figure 1. The tool is able to evaluate the answer with respect to the indications given by the professor and to assign immediately a grade, which is saved and can be provided to the student or not.

3 | METHODOLOGY

The methodology we have used in this research to implement CA is based on the Design Science paradigm for Information Systems research [1, 12, 17]. In Design Science, knowledge and understanding of the problem domain is achieved through the application and development of an artifact, construction, model, or method, starting with a prototype that is modified in successive iterations (instantiations) according to the evaluation performed at the end of each iteration. The development of the prototype allows a deep understanding of the problem in-depth and the feasibility of the proposal to solve it.

The Design Science Research Methodology (DSRM), analyzed by Peffers et al. [22], is used to implement the Design Science paradigm. This methodology describes a process divided into six steps, which are repeated for each iteration except for the first one. We list each step in the first column of Table 1.

TABLE 1 DBR process and its adaptation to this research.

N	Step	Task
1	Identify problem and motivate	The problems of implementing CA have been described in the previous sections of the article. The objective is to prove if an automated assessment tool can be a solution to successfully implement CA in a programming subject.
2	Define objectives of a solution	Define the guidelines to be followed to advance in the CA implementation process and define the measures to evaluate the prototype obtained in the corresponding iteration.
3	Design and development	Design and adapt the automated assessment tool and/or the assessment methodology according to Step 2.
4	Demonstration	Teach the subject using the updated tool and/or methodology as defined in Step 3.
5	Evaluation	Obtain the measures defined in Step 2 and evaluate the implementation prototype.
6	Communication	Share the results of each iteration with other professors and publish them in an article in the final iteration.

Abbreviations: CA, continuous assessment; DBR, design-based research.

For our research, we have chosen the design-based research (DBR) methodology, a variant of DSRM that is appropriate for education. The development of DSRM in the discipline of Information Systems has paralleled that of DBR in educational technology in general and in the development of research in engineering education in particular. In this sense, DBR allows to address the complexity of educational activities under the umbrella of DSRM, which provides the tools to describe and analyze both the design process and the design cycle [5]. Therefore, the DBR follows the same process defined by the DSRM.

The task associated with each step in the context of our research is shown in the second column of Table 1. The description given in Table 1 for the tasks associated with Steps 2 through 6 is general for all iterations. The concrete development of these steps in each of the iterations will be described in the Results section, with the exception of Step 1, which is only included in the first iteration. Although we have already identified the problem and the motivation in the Introduction, we still need to define the goal we want to achieve to finalize the implementation of the CA. In the case of the subject in which we are going to implement the CA, students have to perform different types of activities. Before the CA, the only activities that were assessed were the exams and the subject project, but the rest of them were not assessed at that time, as we will show below. Therefore, we decided that the implementation would end when we could assess all the activities performed by the students.

3.1 | Participants and learning environment

We chose the subject to test the implementation according to the flexibility offered by the professor in charge to modify the assessment methodology. The selected subject is a programming subject in which students learn the Java programming language, including object-oriented programming and web application development. The subject is taught in the second year of the Industrial Management Engineering degree. At our university, the academic year is divided into two 4-month periods, and this subject is taught in the second one. The classes of this subject are held in a classroom where the students have a computer to test their own code and answer the exercises.

The participants of our research were the students of the subject. According to our research design, each iteration of the process corresponds to one academic year. Then, the number of participants in each iteration coincides with the number of students who took the subject in that academic year (Table 2).

Before the introduction of CA, the subject had only four assessments: two midterm exams, a final exam, and an optional project. Each of these assessments include the content taught to students since the beginning of the subject until the previous class to the assessment. These assessments were not removed or changed in time while we implemented new assessments. Then, students were assessed twice or more times about each topic: one

TABLE 2 Number of students per academic year or iteration.

Iteration	Academic year	Students
1	2016–2017	42
2	2017–2018	27
3	2018–2019	42
4	2019–2020	45
5	2020–2021	51

time with the traditional assessments and one or more times with the new assessments.

The course content before the CA included exercises for each topic that students were expected to complete before moving on to the next topic. Students started these exercises in class, with the professor available to help them. They usually did not complete these exercises in class, so they were expected to complete them as homework although they were not graded and did not count toward the final grade.

3.2 | Measurements

As part of the second step of the DBR methodology, we selected three parameters to measure based on the research questions we posed for this study.

First, we stored the final grades of the students to measure their performance with the purpose of checking that it does not decrease as the CA is implemented. The final grade is a summative evaluation that measures student results when applying their knowledge in different stages: immediately with the assessments that take place every week, and after a period with the exams.

The assessments are equivalent from year to year despite the assessments have to change every new course to avoid students copying the answers from past years. The automatic grading tool allows you to duplicate a subject. This feature is used to create the subject outline based on the content created for the previous course, which allows the professor to make only minor changes to the exercises and ensure that the complexity of the assessments is similar. This procedure saves time from the instructor and helps to build a bank of exercises that can be used for other activities.

Second, we saved the grades of the group project that students did throughout the course to verify that their learning was not superficial. The depth of learning that students achieve in the course is reflected in their ability to program and incorporate new features in the group project that they present at the end of the course. The grade of the project is based on a variety of parameters. On the one hand, the originality and the number of modules in the project are evaluated. The students have different project examples, such as the website of an online store with five modules. The originality and the number of modules measure, in a qualitative and quantitative way, the ability of the students to distance themselves from the available examples and to develop new features with the knowledge they have acquired during the course. On the other hand, the oral presentation of the project is assessed, which also allows to assess the participation of each member of the group. All projects from the different courses were graded by the same professors with the same rubrics, so that an objective comparison of all grades can be made.

After collecting the final grades and project grades, we calculated statistics such as the mean of the grades obtained by each course to compare them. We also included in the analysis of these data a Wilcoxon rank-sum test to be able to check if there has been a significant change in the means due to the implementation of CA.

Finally, we surveyed the students to get their opinions about the assessments introduced with the CA. The surveys were administered in the middle of the study period so that students have experience studying with CA. They consisted of up to 14 questions, but we considered only four of them for the study, which gathered the perception of the students about concrete assessments and about the overall assessment methodology. Three of them were multiple-choice questions, and their purpose was to study their study habits regarding the evaluation of different activities in the subject. The fourth question was open-ended, and it asked the students for suggestions that they found interesting about the automated assessment tool and/or the assessment methodology. To analyze the results of the multiple-choice question we gathered the data and we counted the number of students who had chosen each option. In the case of the open-ended question, we use the content analysis methodology. In it, two authors of the study coded the responses obtained and then made a tally of the most mentioned topics.

The measurement of these parameters was carried out in different iterations depending on the changes the professor introduced in the tool and/or methodology, and these data were used to evaluate the corresponding prototype, everything as part of the fifth step of DBR.

4 | RESULTS

As stated in the methodology, Step 1 of the methodology has already been described and it only participates in the first iteration, so we will proceed to describe the development of the rest of the steps in each iteration. Steps 2 and 3 comprise various tasks and their development is described in Table 3. The development of Steps 4, 5, and 6 was repetitive

TABLE 3 Tasks description for Steps 2 and 3 of the DBR.

Academic year	Step 2: Define objectives of a solution	Step 3: Design and development
2016–2017	We will begin the implementation of CA through the automated assessment tool, so we will need to introduce new assessments. Then, student grades should be studied to corroborate the data from other studies and to verify that the implementation of CA does not encourage superficial learning.	We configured two types of exercises in the automated assessment tool. The first were part of a class assessment and counted toward the final grade, while the second were available only for students to practice on their own. The initial setup of automatically graded exercises requires additional time compared to traditional exercises, so we decided that in this iteration there would only be one in-class assessment every 10 days. In addition, the project had been optional in previous years, so we decided to make it mandatory.
2017–2018	Students received better final grades and maintained the grades they received in the project, so we will continue to increase the number of assessments to a weekly frequency. Students should maintain or improve their grades and demonstrate through a survey that they are comfortable with this number of assessments.	We increased the frequency of class assessments to weekly, except during exam weeks and we configured the required exercises in the automated assessment tool. The student survey was conducted in the middle of the subject period and was designed to get information from students about their confidence in learning on their own, their opinion about the usefulness of the assessments introduced, and if they had any suggestions.
2018–2019	The students kept the grades they received in the previous course and the surveys show that they value CA positively, although they suggested having more automatically graded exercises to study on their own. So, we should consider providing more automatically graded exercises for students while we proceed to add the assessment of another activity.	Since students would like to have more exercises graded automatically, we decided to publish more exercises in the tool so that students could do them while studying. It does not take much time to set up these assessments because you can use previously configured exercises with simple modifications. We also specified that students could choose whether or not these exercises would count towards their final grade.
2019–2020	Students kept the grades they received in the previous course, so we validated the assessment of all the exercises in the subject. Then we should proceed to the assessment of the theory.	The automated assessment tool also allows us to configure multiple-choice tests, so we set up questions on the subject concepts to conduct one theory assessment per week. To obtain the time required to perform these assessments, we introduced Flipped Learning so that students have available content to study before class.
2020–2021	Students keep the grades they received in the previous course. The only activity that is not yet assessed in the subject is lecture attendance. In a traditional lecture, there is no way to collect data from students to assess them, so we need to adapt the lectures to include student participation. So, we should check again if the students feel comfortable with a survey, this time with the assessment of all their activities.	To assess this activity, we decided to use live tests, similar to those offered by platforms such as Kahoot. On the one hand, we modified the programming of the tool to allow the live assessment of the theory tests. On the other hand, we adapted the methodology described by Mazur [19]: students have a limited time to answer each test question in groups of 3 or 4, comparing their knowledge. After each question, the teacher is able to visualize the answers and detect gaps in the knowledge of the students, explaining concepts again if necessary.

Abbreviations: CA, continuous assessment; DBR, design-based research.

through the iterations following the description offered in Table 1. We show in Table 4 a schema of the assessments included in the subject as part of Step 4, as well as the measures taken as part of Step 5.

Since the objective of the implementation is to be able to evaluate a greater number of activities that students perform, it was decided to maintain the previously existing evaluations with the same structure in all

TABLE 4 Assessments and measures used for Steps 4 and 5 of the DBR.

	Step 4: Demonstration							Step 5: Evaluation		
Academic year	Midterm exams	Project	Final exam	Class exercises	Studying exercises	Studying theory	Professor explanations	Final grade	Project grade	Survey
Before CA	2	Optional	Yes	No	No	No	No	Yes	Yes	–
2016–2017	2	Yes	Yes	Every 10 days	No	No	No	Yes	Yes	No
2017–2018	2	Yes	Yes	Weekly	No	No	No	Yes	Yes	Yes
2018–2019	2	Yes	Yes	Weekly	Optional	No	No	Yes	Yes	No
2019–2020	2	Yes	Yes	Weekly	Optional	Weekly	No	Yes	Yes	No
2020–2021	2	Yes	Yes	Weekly	Optional	Weekly	Weekly	Yes	Yes	Yes

Abbreviations: CA, continuous assessment; DBR, design-based research.

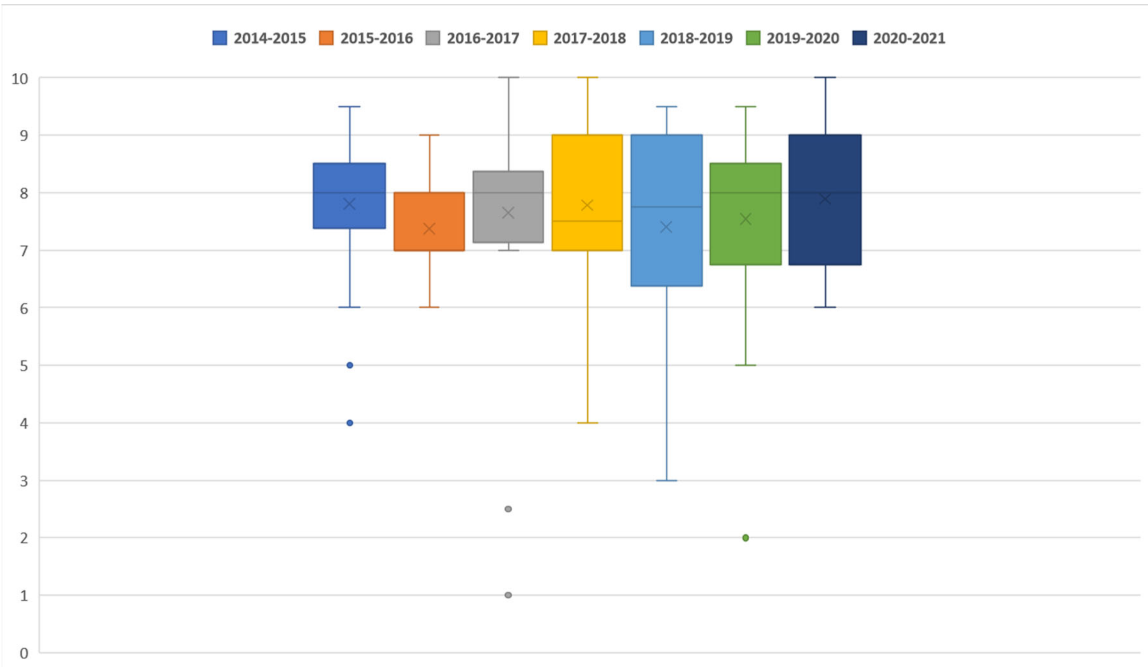


FIGURE 3 Boxplot of the grades of the projects grouped by academic year.

iterations. These evaluations not only allow us to assess the knowledge of the students over time and in a grouped manner but also allow us to make a comparison with the years before the implementation of the CA.

4.1 | Grades of the students

At our university, we grade students on a scale of 0–10, and students pass the subject if they receive at least a 5. In the case of the project grades, we compare them using box plots (Figure 3), only considering the projects that were submitted and presented. In the case of the final grade, we grouped the students into six different grade ranges to compare the results across years: (0–5), (5–7), (7–9), (9-10), and those who have decided to drop the subject (Figure 4). Since the number of students in each course varies, we show the percentage of the class that belonged to each group. In both figures, data from two academic years before the implementation of CA (2014–2015 and 2015–2016) are shown to provide information about the previous situation of the subject.

Meanwhile, we performed Wilcoxon rank-sum tests between the courses 2014–2015 and 2020–2021. The results are shown in Table 5. On the one hand, in the case of the final grades, the *p*-value is less than .05, therefore the mean of these grades is significantly different after the introduction of the CA. On the other hand, the *p*-value obtained from the

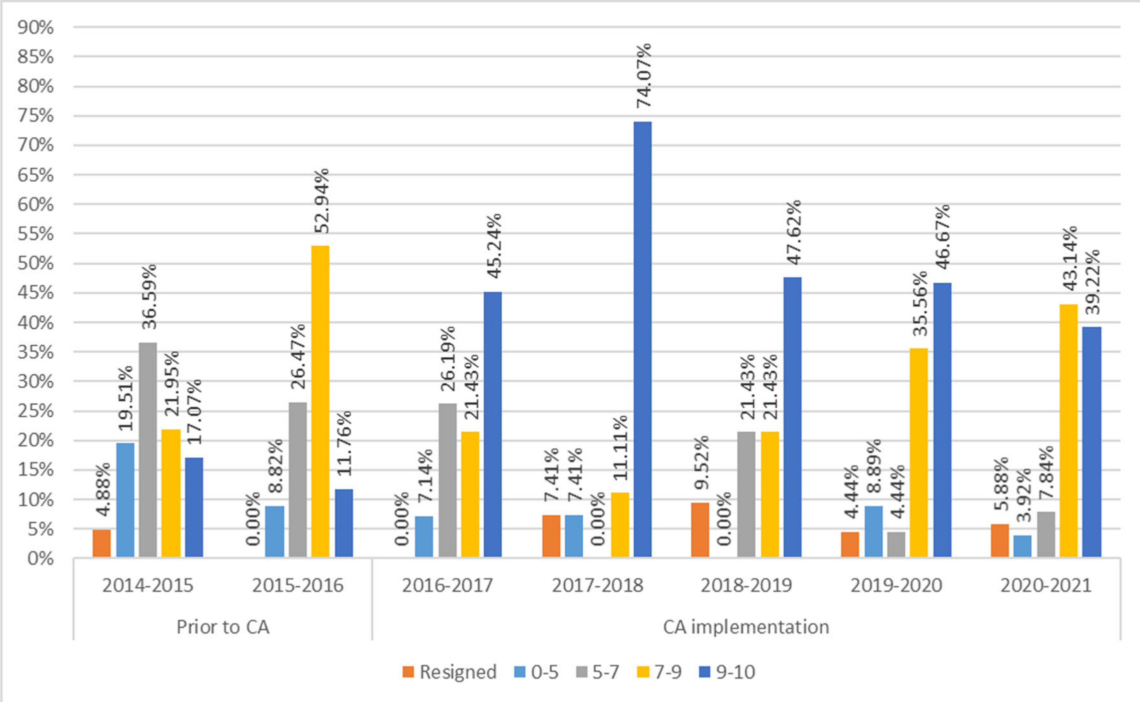


FIGURE 4 Evolution of the ranks of grades of the students in each academic year.

TABLE 5 Wilcoxon rank-sum tests results.

Grades	*p*-Value
Final grade	6.446e−07
Project	.3675

TABLE 6 Number and percentage of participants per survey.

Survey	Academic year	Participants	% Of the class
1	2017–2018	23	85.19
2	2020–2021	38	74.51

test carried out with the project grades is not conclusive, so we could guess that the grades are similar before and after the CA implementation.

4.2 | Surveys

During the implementation we conducted two surveys in the 2017–2018 and 2020–2021 academic years, corresponding to the second and final iterations. The number of participants and the percentage of participation in the surveys are shown in Table 6, while the results of the three multiple-choice questions are shown in Figure 5.

In the case of the open-ended question, the content analysis showed that the most common comment in the surveys was the request for more exercises assessed through the assessment tool. The students expressed in these comments a preference for the exercises to be published in the tool because they were graded automatically and received feedback in the moment.

5 | DISCUSSION

5.1 | Grades of the students

Throughout the development of the CA implementation, we have stored the project grade and the final grade of students. The project grades did not vary significantly over the years, so we could verify that the learning of the students was not superficial. For example, both the mean and the median have remained in a range between 7 and 8 points, as can be seen in Figure 3. In addition, the Wilcox rank-sum test between the first and last studied years showed that the means of those years are not statistically different.

On the contrary, the final grades were not constant throughout the courses as Figure 4 shows us and we could notice interesting trends through the years. Before implementation of CA, most students were in the 5–7 point and 7–9 points ranges. Whereas in the later years, the majority of students were in the 7–9 and 9–10 points ranges. In particular, it appears that the percentage of the 5–7 points group decreases in subsequent iterations. This

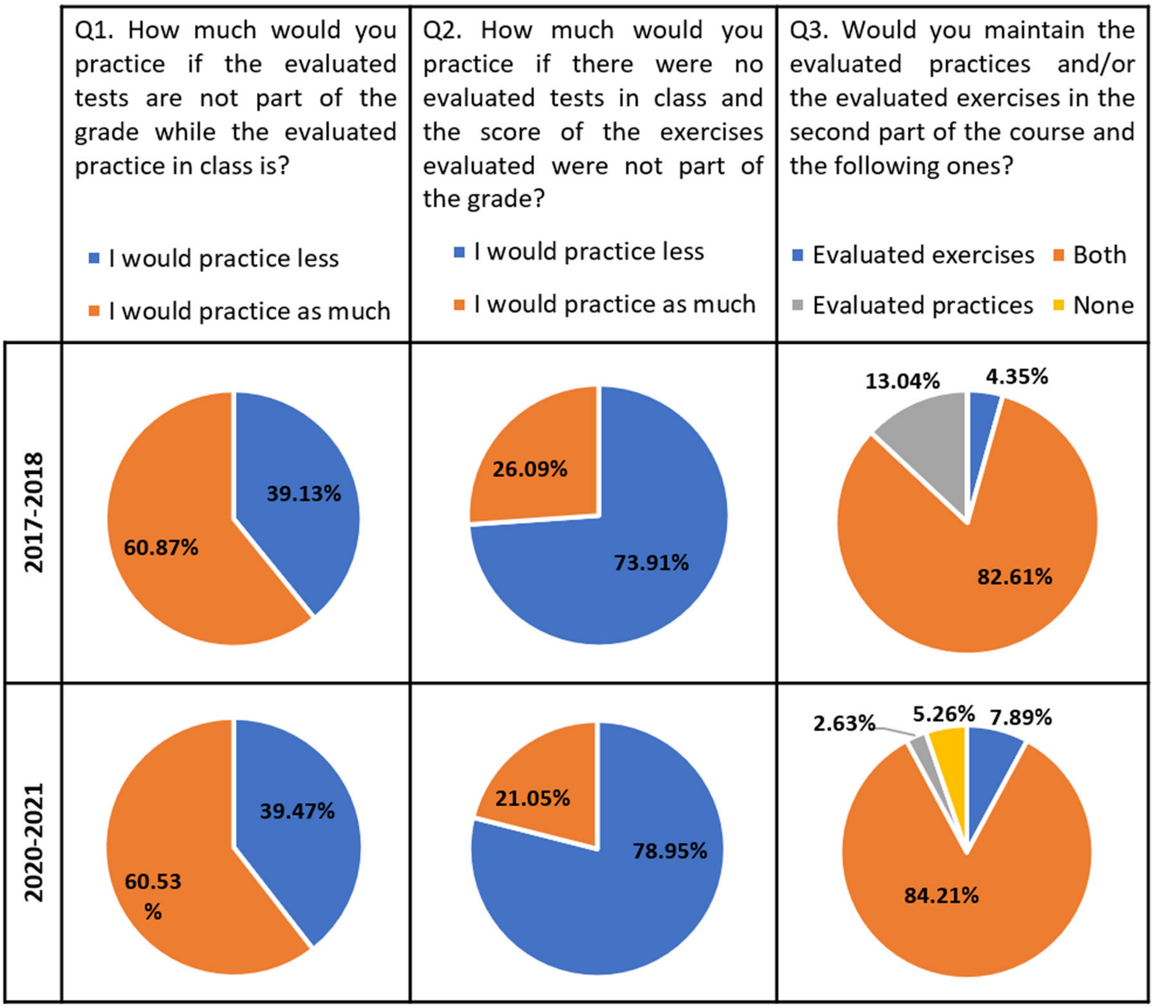


FIGURE 5 Results of the surveys.

may indicate that the type of student who used to pass with a low grade between 5 and 7 points is getting a grade above 7 points after the CA implementation.

In the study conducted by Ferreira et al. [9], it is shown that there is no significant improvement between students who use and those who do not use CA. This difference with our study could explained by the fact that we have applied successive changes in the methodology which gradually improved the performance of the students, since the difference between the results before and after the implementation of CA is greater and greater as we progress through the iterations.

Another noteworthy trend is that in the first iteration, the majority group became the 9–10 points group and maintained its percentage in subsequent iterations. However, since the second iteration, the percentage of the group in the 7–9 points range has increased each year until it surpassed the 9–10 points group in the last iteration. Therefore, it is possible that the assessments introduced in the early iterations are helping one group of students achieve the highest scores. While the assessments introduced in the later iterations help another group of students to achieve grades between 7 and 9 points.

Lastly, we have observed that since the second iteration, a small group of students has been continuously dropping out of the course. This behavior did not exist in previous years, when students dropped the course sporadically. The same trend can be found in other studies like the one by Montolio [21], who concludes that the number of students assessed at the end of the subject was reduced after the implementation of CA.

5.2 | Surveys

We wanted to measure with two of the multiple-choice questions how confident students would be if we removed the additional assessments, which is equivalent to coming back to the methodology used previously for the implementation. In the first of these questions, we asked what would happen if we removed the automatically graded exercises that students were doing to study on their own. The majority of students, about 60%, said in both surveys that their study habits would not be affected in this case. In the second question, we also considered removing the graded class exercises that were done on a weekly basis. The answers of the students changed in both surveys: most of the students said that they would study less on their own, about 75%. The goal with the remaining multiple-choice questions was to assess how much students valued these new assessments. In both surveys, more than 80% of the students wanted to keep both assessments. The rest of the students chose to keep one of the two assessments, while there were just two students, both from the second survey, who would prefer to remove both assessments.

These answers indicate that most students value these assessments because they feel more motivated to study more than they would if they were doing it on their own. However, the surveys do not provide clear information about student preferences for any of the assessments. At the same time, the appearance of students that prefer to remove the assessments in the second survey may indicate that they may feel uncomfortable with the increased number of assessments.

The open-ended question of the survey asked students for their suggestions. Their most common answer was that they wanted more exercises in the tool. They would like to have additional exercises to the ones presented in the assessments. In addition, some of them expressed that they tend not to look at the exercises outside of the tool. As part of the subject, the exercises available to students in the automated assessment tool are those from the assessments. However, they have more published exercises in the course documentation to practice on their own. These suggestions show that students prefer the automated assessment to the point that once they get used to this type of assessment, they may find it difficult to study with traditional exercises. This observation is very similar to the one obtained in the study conducted by Restrepo-Calle et al. [26] after interviewing students about the use of automated assessment. In the interview, the students suggested that the assessment tool should be constantly available.

5.3 | Additional findings

In addition to the information obtained from the data collected, the development of the study has shown us relevant outcomes that we had not contemplated at the beginning of this study:

First, having control over the development of the automated assessment tool gave us tremendous flexibility in designing the CA. We were able to implement any innovation in either or both areas seamlessly and without restriction. As a result, the successive changes introduced in each iteration did not cause us any problems beyond the time it took to configure the automatically graded exercises and update the tool.

The professor can perform different assessments (code correction, multiple choice questions, live tests for formative evaluation in class, etc.) using the same tool, which simplifies data collection. Both professors and students would not have to constantly switch between different platforms to retrieve data or complete exercises. Any assessment methodology should provide professors with the necessary information to make appropriate decisions and allow students to reach the desired level of knowledge [11]. Therefore, if data generated by CA provide more information than other assessment methods, professors and students are more likely to achieve these goals. The automated assessment tool used in this study collects a significant amount of data: more than 400 grades per student. These data could be used in a future study to obtain information about the evolution of learning in the class.

The automated assessment tool helps students to develop additional skills. On the one hand, the tool grades the answer only if the code introduced can be compiled. Thus, students get used to developing programs according to an incremental model: they test a functional iteration until it works before adding new functionality. On the other hand, automated grading encourages students to debug their own code from the beginning of the course, as they do not have to wait for the assessment of the professor.

6 | CONCLUSION

CA is an assessment methodology that consists of constantly evaluating students in the different activities they perform to acquire knowledge. The problem with this methodology is that it increases the amount of time that teachers spend on assessment. On the other hand, if the methodology is implemented incorrectly, it can promote superficial learning among students. To solve this problem, we have implemented continuous evaluation in a programming subject using an automated evaluation tool developed by our research group. This implementation was carried out through the DBR methodology, which indicates a series of steps to be followed in different iterations, in which a prototype is obtained. In this study, the prototype consists of the resulting automated evaluation tool in conjunction with the proposed evaluation methodology. The goal was to develop a prototype that could assess all the activities that students do in this subject. The prototype was achieved after five iterations in which we added the assessment of up to four subject activities. In each iteration, we considered student performance and depth of knowledge. And in two of them, we also considered student feedback. With this data, we can answer the research questions posed in the introduction of this study.

First, we measured the performance of students each year based on the final grades they received in the subject. We found that the final grades of the students improved after the implementation of CA. The data show that the group of students who previously passed with a minimum grade (5–7 points) decreased, while the two groups with a grade higher than 7 increased. This could indicate that the introduced assessments help the types of students with a final grade between 5 and 7 points and between 7 and 9 points to move to a grade corresponding to the group immediately above. It seems that the theory assessments help the first of these groups, while the practice assessments help the second more.

In the years before the implementation of the CA, the dropouts occurred towards the end of the course and were quite sporadic. After the implementation, it has been observed that the group of students who drop out is consistent in all courses and that the time of dropping out is distributed throughout the period of the subject, not only before the final exams. This behavior could indicate that CA helps students become more aware of their level of knowledge in the subject. Thus, further interviews could confirm whether these students drop because they know they will not be able to pass the subject at the end of the semester.

Second, we recorded the students' grades received for submitting and presenting the project they completed in the course to verify that CA did not promote superficial learning of the concepts taught. The statistical data of the submitted projects do not show a significant difference between the different courses. Therefore, we can affirm that the learning of the students was not more superficial after the implementation of CA, although it was not deeper either. However, the professor of the course has stated that it seems that the quality of the projects has improved in recent years, so another study could be carried out with different measures to contrast them with the data obtained in this study.

Third, considering the student views, the survey responses show that they prefer to have automatically graded exercises and they ask for more available exercises in the automatic grading tool. Therefore, we can state that not only they have no problems against CA, but also the majority of them would like to continue using CA because it motivates them to have better study habits and as a result they are more involved in the subject. Students may value CA because the immediacy of the feedback motivates them to complete more exercises because they spend less time checking their answers. However, we should consider in further studies that they may become so accustomed to this assessment that they spend less time reflecting on their mistakes or checking their answers. It could even affect their motivation to study other subjects in which they do not have this method.

This study also provided new insights into the use of a customized automated assessment tool that may point to new lines of research. In general, the tool provides great flexibility to adapt to the teaching methodology used for teaching and allows the collection of a large amount of data about students that can be analyzed in the future. And in particular, in the teaching of programming, it encourages students to use the incremental development model.

Before concluding, we must make a brief reference to the COVID-19 pandemic, a major educational event that occurred during the period covered by this study. Like most educational institutions in many countries, our university stopped offering face-to-face classes and switched to distance education in March 2020. In this course, as shown in Tables 3 and 4, we introduced multiple-choice tests for the assessment of theory in the subject of this study, in addition to the introduction of flipped learning. This decision made it possible to easily continue the development and assessment of the subject after the change to distance education [3].

Finally, as future work, we should repeat this implementation in other subjects to verify that the tool is also suitable to implement CA in a greater number of situations and contexts. In this sense, as already mentioned, the flexibility of the approach gives us great hope that it will be possible to repeat this experience with satisfactory results.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Kevin Calderon https://orcid.org/0000-0001-8743-5063

REFERENCES

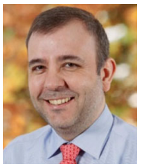
1. J. E. van Aken, *Management research as a design science: articulating the research products of mode 2 knowledge production in management*, Br. J. Manage. **16** (2005), 19–36.
2. S. Bloxham and P. Boyd, *Developing effective assessment in higher education: A practical guide*, McGraw-Hill International, New York, NY, 2007.
3. K. Calderon, C. Blanco, I. Gutierrez, N. Serrano, and J. Santos, *Evaluation of the emergency remote teaching during Covid-19 lockdown in a Spanish University*, J. Univ. Teach. Learn. Pract. **19** (2023).
4. D. Carless, G. Joughin, and N.-F Liu, *How assessment supports learning: Learning-oriented assessment in action*, Hong Kong University Press, Hong Kong, 2006.
5. A. K. Carstensen and J. Bernhard, *Design science research—A powerful tool for improving methods in engineering education research*, Eur. J. Eng. Educ. **44** (2019), 85–102.
6. J. S. Cole and S. W. T. Spence, *Using continuous assessment to promote student engagement in a large class*, Eur. J. Eng. Educ. **37** (2012), 508–525.
7. S. W. Draper, *Catalytic assessment: Understanding how MCQs and EVS can foster deep learning*, Br. J. Educ. Technol. **40** (2009), 285–293.
8. K. Edström, M. Engström, and Å. Wiklund, *Student involvement in principled change: Understanding the student experience*, 2003.
9. D. J. Ferreira, H. C. da Silva, T. F. N. Melo, and A. P. Ambrósio, *Investigation of continuous assessment of correctness in introductory programming*, J. Educ. Technol. Soc. **20** (2017), 182–194.
10. G. R. Gibbs and C. Simpson, *Conditions under which assessment supports students' learning*, Learn. Teach. High. Educ. **1** (2005), 3–31.
11. J. W. Gikandi, D. Morrow, and N. E. Davis, *Online formative assessment in higher education: A review of the literature*, Comput. Educ. **57** (2011), 2333–2351.
12. A. R. Hevner, S. T. March, J. Park, and S. Ram, *Design science in information systems research*, MIS Quart. **28** (2004), 75–105.
13. J. Heywood, *Engineering education: Research and development in curriculum and instruction*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2006, pp. 3–18.
14. S. Jordan, *Assessment for learning: Pushing the boundaries of computer-based assessment*, Pract. Res. High. Educ. **3** (2009), no. 1, 11–19.
15. M. J. Laakso, E. Kaila, and T. Rajala, *ViLLE—Collaborative education tool: Designing and utilizing an exercise-based learning environment*, Educ. Inform. Technol. **23** (2018), 1655–1676.
16. S. F. Leung, E. Mok, and D. Wong, *The impact of assessment methods on the learning of nursing students*, Nurse Educ. Today **28** (2008), 711–719.
17. S. T. March and V. C. Storey, *Design science in the information systems discipline: An introduction to the special issue on design science research*, MIS Quart. **32** (2008), 725–730.
18. F. J. Martín-Carrasco, A. Granados, D. Santillán, and L. Mediero, *Continuous assessment in civil engineering education: Yes, but with some conditions*, Proc. Int. Conf. Comput. Support. Educ. **2** (2014), 103–109.
19. E. Mazur, *Farewell, lecture?* Science **323** (2009), 50–51.
20. M. Misut and M. Misutova, *Software solution improving productivity and quality for big volume students' group assessment process*, Int. J. Emerg. Technol. Learn. **12** (2017), 175–190.
21. D. Montolio, *The impact of continuous assessment on a temporal perspective: The results of a pioneering experiment at the University of Barcelona (Spain)*, Multidiscip. J. Educ. Soc. Technol. Sci. **2** (2015), 128–140.
22. K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, *A design science research methodology for information systems research*, J. Manage. Inform. Syst. **24** (2007), 45–77.
23. J. E. Pérez-Martínez, M. J. García-García, W. H. Perdomo, and M. J. Villamide-Díaz *Analysis of the results of the continuous assessment in the adaptation of the Universidad Politécnica de Madrid to the European higher education area*, Proc. Res. Eng. Educ. Symp., 2009, Palm Cove, QLD.
24. P. S. Pietikäinen, and R. S. Karinen, *Continuous assessment as a tool for deeper learning, for Engineering Education*, Tampere, Finland, 2016.
25. M. D. Reina-Paz, A. Rodriguez-Oromendia, and C. Sevilla-Sevilla, *Effect of continuous assessment tests on overall student performance in the case of the Spanish National Distance Education University (UNED)*, J. Int. Educ. Res. **10** (2014), 61–68.
26. F. Restrepo-Calle, J. J. Ramírez Echeverry, and F. A. González, *Continuous assessment in a computer programming course supported by a software tool*, Comput. Appl. Eng. Educ. **27** (2019), 80–89.
27. C. Sangwin, Computer aided assessment of mathematics, Oxford University Press, Oxford, 2013.
28. N. Serrano, C. Blanco, K. Calderón, I. Gutierrez, and M. Serrano, *Continuous assessment with flipped learning and automated assessment*, Proc. 17th Int. CDIO Conf., 2021.
29. N. Serrano, C. Blanco, F. Carias, and E. Reina, *Information from automated evaluation in an engineering school*, Fourth Int. Conf. High. Educ. Adv., 2018.
30. C. M. Tan, *An evaluation of the use of continuous assessment in the teaching of physiology*, High. Educ. **23** (1992), 255–272.
31. R. X. Thambusamy and P. Singh, *Online assessment: How effectively do they measure student learning at the tertiary level?* Eur. J. Soc. Behav. Sci. **30** (2021), 63–76.
32. E. Trotter, *Student perceptions of continuous summative assessment*, Assess. Eval. High. Educ. **31** (2006), 505–521.

AUTHOR BIOGRAPHIES

Kevin Calderon is a PhD student graduated in Telecommunication Engineering in the School of Engineering of the University of Navarra, Spain. His research is focused on the development and application of data analytics techniques in the educational and learning area.

Nicolás Serrano is a professor of Languages and Computer Systems at the School of Engineering of the University of Navarra, in San Sebastian. He has worked at the CEIT research center and in industrial and service companies. His experience covers the areas Computer Science, Software Engineering, Digital Technology and Information Systems. He has directed and developed several information systems such as the academic management system of the University of Navarra, an ERP in web platform for business management and applications for technological innovation.

Carmen Blanco is an associate professor of Mathematics, School of Engineering of the University of Navarra (Tecnun), Spain. Currently, her research is focused on the analysis, development and application of the new technologies and methodologies for the improvement of the learning process.



Iñigo Gutierrez is professor of Electronic Technology and Director of Development at the School of Engineering of the University of Navarra (Tecnun), Spain. Currently, his research is focused on the analysis and application of new teaching technologies and methodologies.

How to cite this article: K. Calderon, N. Serrano, C. Blanco, and I. Gutierrez, *Automated and continuous assessment implementation in a programming course*, Comput. Appl. Eng. Educ. **32**, (2024), e22681. https://doi.org/10.1002/cae.22681