

# Learning Analytics to Detect Evidence of Fraudulent Behaviour in Online Examinations

Antonio Balderas<sup>1\*</sup>, Manuel Palomo-Duarte<sup>1</sup>, Juan Antonio Caballero-Hernández<sup>2</sup>, Mercedes Rodriguez-Garcia<sup>3</sup>, Juan Manuel Doderó<sup>1</sup>

<sup>1</sup> Departamento de Ingeniería Informática, Universidad de Cádiz, Escuela Superior de Ingeniería, Puerto Real (Spain)

<sup>2</sup> EVALfor Research Group, Universidad de Cádiz, Puerto Real (Spain)

<sup>3</sup> Departamento Ingeniería en Automática, Electrónica, Arquitectura y Redes de Computadores, Universidad de Cádiz, Escuela Superior de Ingeniería, Puerto Real (Spain)

Received 26 February 2021 | Accepted 2 September 2021 | Published 27 October 2021



## ABSTRACT

Lecturers are often reluctant to set examinations online because of the potential problems of fraudulent behaviour from their students. This concern has increased during the coronavirus pandemic because courses that were previously designed to be taken face-to-face have to be conducted online. The courses have had to be redesigned, including seminars, laboratory sessions and evaluation activities. This has brought lecturers and students into conflict because, according to the students, the activities and examinations that have been redesigned to avoid cheating are also harder. The lecturers' concern is that students can collaborate in taking examinations that must be taken individually without the lecturers being able to do anything to prevent it, i.e. fraudulent collaboration. This research proposes a process model to obtain evidence of students who attempt to fraudulently collaborate, based on the information in the learning environment logs. It is automated in a software tool that checks how the students took the examinations and the grades that they obtained. It is applied in a case study with more than 100 undergraduate students. The results are positive and its use allowed lecturers to detect evidence of fraudulent collaboration by several clusters of students from their submission timestamps and the grades obtained.

## KEYWORDS

Cheating, Evaluation, Learning Analytics, Learning Management System, Learning Records.

DOI: 10.9781/ijimai.2021.10.007

## I. INTRODUCTION

THE learning process in higher education is no longer conceivable without information technology. In particular, Learning Management Systems (LMSs) are a meeting point for students and faculties in the university, where faculties organise their courses and set up learning activities, while students find learning material and communicate with their lecturers. However, in terms of student assessment, lecturers usually prefer face-to-face rather than online examinations. The main reason for this is the concern that students can easily cheat in online examinations because the lecturers lose control of what the students do while taking their examinations [1].

This concern has become particularly significant in the context of the Covid-19 pandemic. Due to forced confinement in spring 2020, all Spanish students stopped attending classes and lectures, and moved from classrooms to video conferencing [2]. The same applied to evaluation activities, which were moved to the LMS through individual assignments, questionnaires and synchronous oral interviews, among others [3]. Oral examinations can be a solution to ensure the absence of fraud in a student's examination. Unfortunately, oral examinations

are not always possible, either because they are not sustainable when there are too many students or because the course matter is not suitable for oral communication. On the other hand, examinations based on multiple-choice questionnaires were widely used as they are immediate to grade through their automated settings. However, they are prone to cheating by students [4].

To alleviate their concerns about student cheating during the confinement, lecturers could propose different examinations that were harder than those they usually set in face-to-face sessions (e.g., more difficult multiple-choice questions, shorter time to answer, etc.). In addition, the students sometimes reported that the LMS often suffered from connectivity problems because of the number of simultaneous connections taking place during the examination. These problems led to interruptions in the LMS service during the examinations, which would be a serious problem for the students involved. Because this issue was out of the lecturer's control, the students asked for the examinations to be asynchronous to alleviate this problem. Asynchronous examinations allow students to take the examination at different times [5].

This research focuses on the detection of cheating behaviour of students when asynchronously taking exams or submitting assignments. To get an insight into students who cheat in examinations, lecturers can check the student records on the LMS. Unfortunately, the information provided for large groups of students

\* Corresponding author.

E-mail address: antonio.balderas@uca.es

is hard to manually analyse. Learning analytics support lecturers in both improving the assessment of their students and monitoring their learning process [6], [7].

The research question that arises from this context is: Can lecturers collect evidence of cheating students through LMS activity records? This study applies learning analytics techniques to help detect evidence of students who fraudulently collaborate in online examinations. We propose a process model to obtain evidence of how the students take examinations based on their submission timestamps and the grades obtained. For this purpose, we developed Py-Cheat, which is a software tool to identify evidence of fraudulent collaboration among students when performing online activities. It is applied in a case study with more than 100 undergraduate students who submitted several programming assignments throughout the course and later took an examination based on a multiple-choice questionnaire.

The rest of the paper is structured as follows. In the second section, we describe the background of this work. In the third section, we present the materials and methods used, including the Py-Cheat tool. The fourth section presents the results. The fifth section discusses the implications of this study. Finally, the conclusions of this study are drawn in the last section.

## II. BACKGROUND

Although students know and recognise that cheating during their examinations is an ethically unacceptable behaviour, most of them admit to having done it at some point during their academic years [8]. According to Albrecht [9], three circumstances must be present for the student to be driven to cheat: some sort of pressure, the possibility of not getting caught, and the ability to rationalise the action as acceptable.

Concerning being under some sort of pressure, final examinations are, by nature, stressful and a source of anxiety for students [10]. Moreover, the personal and family concerns of living through a pandemic can exacerbate both pressure and stress [11], [12].

Lecturers are blind to what each student is doing while taking the online exam; for example, whether they are accompanied by someone who can help them in the examination, whether they are taking the examination collaboratively with other students via online media, or even whether they look up course materials that they should not consult [13]. The lack of control in online assessment encourages students who "massively copy and plagiarise" anyway, to do it more and more often; and even more in the case of multiple-choice questionnaires, where it is easier to cheat [14].

Setting multiple-choice questionnaires based on the random selection of questions from a pool makes it more difficult to share content as fewer questions are repeated among students' exams. However, if the question pool is used repeatedly in later editions of the same examination, most of the questions become known to the latest students to take the exam. [15].

In computer programming courses, the difference between cheating and collaboration is a bit unclear [16]. Computer engineering students often undertake pair programming assignments, and this partnership, which began with pairs of students handing in assignments, may turn into fraudulent collaborations on examinations [17].

Tools are available to detect cheating by students (i.e., plagiarism detection systems); for example, Turnitin and Viper are the most widely used in higher education [18]. Although they are effective tools, they are hardly applicable to multiple-choice questionnaires and are more focused on looking for semantic similarities between sentences and between words, which is more suitable to detect unfair practice in projects or memorandums [19].

E-proctoring tools have become popular in the pandemic context and they have been used by some educational institutions to detect fraud in online examinations [20]. By using these tools, lecturers can remotely monitor students while they are taking the examination. In this research, conducted with computer engineering students, the lecturers concluded that their students cheated on online examinations, as they found significant differences between the grades of proctored and non-proctored students [21]. Unfortunately, institutions have to provide a tool to their teaching staff, who are reluctant to implement e-proctoring because of the security and privacy issues that it entails [22].

In contrast to such preventive strategies, learning analytics allow lecturers to collect evidence of the students' work in an LMS to implement pre-emptive countermeasures for cheating. Previous work has demonstrated the effectiveness of these evidence-gathering techniques when used to assess individual student performance on skills or learning outcomes [23]–[25]. In other works, researchers have collected evidence of collaboration between students performing an assignment in a group [26], [27].

Based on previous work on evidence collection from LMS activity records, this research aims to collect evidence to detect unfair collaborations that the students should not make during online examinations based on their submission timestamps and the grades obtained.

Most of the studies in the literature that evaluate using multiple-choice questionnaires focus on preventing student cheating before it occurs [28], but not on detecting it afterwards. Several of the proposals used in computer engineering courses for this aim are based on software that automatically creates customised questionnaires [29], [30]. In this way, any two students' questionnaires will be different and it will be more difficult for them to benefit from sharing their content.

In a recent work [31], researchers develop an intelligent agent that tries to anticipate fraudulent student behaviour in real-time. The agent uses both IP addresses and time of response to questions to detect suspicious patterns of behaviour.

Finally, Jaramillo et al. present an algorithm to detect students who collaborate fraudulently by sharing questions/answers. The algorithm is based on submission timestamps and exam responses [32].

## III. MATERIALS AND METHODS

This research proposes a method to detect students' cheating behaviour when taking examinations on LMS based on their submission timestamps and the grades obtained. This method is presented as a process model that uses the LMS activity records to identify evidence of fraudulent collaboration.

### A. Model for Cheating Detection

Fig. 1 shows the model that we proposed for detecting evidence of fraudulent collaboration during students' examinations. The implementation of the model requires, first, an LMS to create examinations or assignments. Second, the LMS must allow lecturers to access and download the activity records. Finally, a software tool such as Py-Cheat is required to process the learning records and look for evidence of cheating. The model consists of a series of steps, as described below:

1. Design assessment instrument: the lecturer designs the examination on the LMS following the course syllabus.
2. Taking assessment instrument: students complete the task required in the assessment instrument.
3. Collect LMS records: the lecturer downloads the information from the LMS activity records.

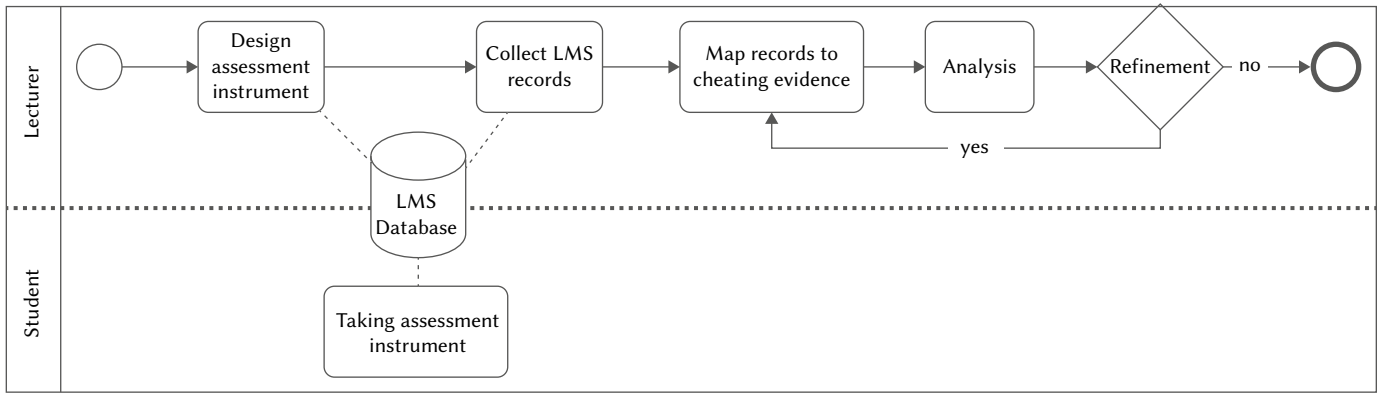


Fig. 1. Model for detecting evidence of cheating on student examinations.

Time	User's full name	Affected user	Activity	Component	Event name	Description
1 February 2021, 19:08	Student 1	Student 1	Questionnaire: Final Exam	Questionnaire	Attempt submitted	The user with id '4051' has submitted the attempt with id '154251' for the quiz with course module id '141931'.
1 February 2021, 18:10	Student 1	Student 1	Questionnaire: Final Exam	Questionnaire	The attempt has begun	The user with id '4051' has started the attempt with id '154251' for the quiz with course module id '141931'.

Fig. 2. Example of the start and end records of a student's completion of a questionnaire.

- Map records to cheating evidence: the software tool supports the lecturer in mapping the students' records to different behaviours that can be evidence of fraudulent collaboration.
- Analysis: the lecturer analyses the evidence and compares it with their observations of the course.
- Refinement: the lecturer can finish the process or refine the evidence, either because they discard the previous ones or to reinforce those that were previously collected.

### B. LMS Activity Records

From the information that can be found in the LMS activity records regarding student's action, this method requires the following:

- Timestamp (*T*): timestamp at which the action was carried out.
- Student (*S*): student who carried out the action.
- Activity (*A*): activity in which the student participated.
- Event (*E*): action performed by the student (e.g., access, respond, submit, etc.)

This information is enriched with the grade (*G*) obtained by the student in the examination and the laboratory group (*L*) to which the student belongs.

### C. Evidence of Cheating

The method requires two events of each student's online examination completion to provide the evidence: the start time (*ST*) and the finish time (*FT*). This information is obtained from the processing of the LMS activity records. In the Fig. 2, we can see an example of the start and finish records for an activity. In this example, the activity is the "Questionnaire: Final exam". Its start time is February 1<sup>st</sup>, 18:10 (*ST* = 18:10 2021-02-01), taken from the event "The attempt has begun". Its submission time is February 1<sup>st</sup>, 19:08 (*FT* = 19:08 2021-02-01), taken from the event "Attempt submitted". Therefore, the completion time (*T*) is 58 minutes (*T* = *FT* - *ST*).

Thus, this method returns sets of students who took the examination or who submitted the assignment sequentially and probably collaborated on it. To detect the collaboration, the method is based on the values of three features of two students' examinations.

Given two students:

$$S_1 \rightarrow \{L_1, ST_1, FT_1, T_1, G_1\}$$

$$S_2 \rightarrow \{L_2, ST_2, FT_2, T_2, G_2\}$$

$S_1$  and  $S_2$  are considered to have collaborated in carrying out an activity if:

- $S_2$  starts the examination right after  $S_1$  submits it within a time interval (*I*) defined by the lecturer:  $FT_1 \leq ST_2$  and  $ST_2 - FT_1 = I$
- $S_2$  improves the grade/minutes ratio with respect to  $S_1$ :  $G_2/T_2 = G_1/T_1$   
This evidence arises from two observations:
  - $S_2$  usually takes the same or less time to complete the examination than  $S_1$ :  $T_2 \leq T_1$
  - $S_2$  usually achieves a grade equal to or greater than  $S_1$ :  $G_1 \leq G_2$
- Additionally, the lecturer can configure the results to check if the students belong to the same laboratory group:  $L_1 = L_2$

Fig. 3 shows an example. If the method returns a cluster of five students ( $S_1, S_2, S_3, S_4$  and  $S_5$ ), this means that:

- Students completed the exam sequentially. While student  $S_1$  (examinee) took the exam through the LMS, the students  $S_2, S_3, S_4$  and  $S_5$  (collaborators) helped  $S_1$  to solve it.
- Sequentially, the roles of examinee and collaborators changed until the five exams were completed.
- The last members of a sequence are likely to get higher grades. Firstly, because they have been able to repeat exam questions and, secondly, because they have had more time to search for the answers.

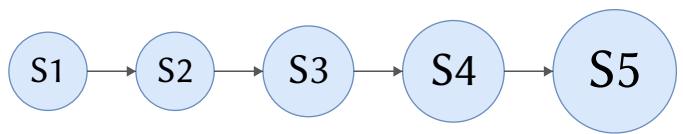


Fig. 3. Example of a cluster with five students (represented by nodes) who took the examination sequentially and helped each other in solving the exam. The diameter of the nodes symbolises the student's grade. The later they take the examination, the higher grade they get.

#### D. Py-Cheat Tool

Py-Cheat<sup>1</sup> is an open-source software tool that implements our proposal to detect students' fraudulent collaboration on activities based on the LMS activity records. The objective of Py-Cheat is to convert the collected records into evidence that can be more easily analysed by the lecturer.

Py-Cheat requires the lecturer to provide a CSV file with the following information: student (*S*), laboratory group (*L*), start time of the activity (*ST*), submission time of the activity (*FT*), completion time (*T*) and grade (*G*).

Once the lecturer provides the CSV file, Py-Cheat requests the lecturer to set the following parameters:

- Interval (*I*): maximum time that can elapse between two students submitting their assignment consecutively to be considered suspicious of fraudulent collaboration.
- Minimum number of students: the minimum number of students to check for fraudulent collaborations.

Py-Cheat processes two types of activities: assignments and questionnaires. In both cases, Py-Cheat returns a set of students who have made sequential submissions according to the specified criteria. However, there is a difference for assignments because, as opposed to questionnaire examinations, it does not take into account the duration. This happens because the start date of an examination is specifically defined in the record for each student (i.e., when the student clicks the start examination button). However, the start time of an assignment is common to all, and the LMS only records the time at which students submit the assignment. For the interval between two students in a sequence when submitting an assignment, Py-Cheat considers the difference between the submission dates:  $FT_2 - FT_1 \leq I$ .

Finally, Py-Cheat also provides a directed graph that represents a network of students who collaborated in the examination or the assignment according to the specified criteria.

### IV. CASE STUDY

The participants are 132 students from the University of Cadiz (Spain) who were enrolled in Databases, which is a second-year compulsory course on the Computer Engineering Degree during the second semester of the 2019-20 academic semester.

In this university, the LMS is based on Moodle. We used Py-Cheat to analyse the activity records of the following activities:

- Questionnaire: a 10-multiple-choice examination corresponding to the final SQL language practice examination of the course.
- Assignments: five SQL-language practical assignments that students had to submit during the semester.

#### A. Questionnaire

Based on the students' complaints, as mentioned in the introduction, the examination was configured asynchronously to avoid possible problems of LMS downtime. Similarly, students were given 2 minutes and 30 seconds to answer each question, which addressed their complaint about the limited time that they had in other courses.

- A total of 10 multiple-choice questions were presented in sequence (return to previous questions was not allowed).
- Once started, the examination could not be paused.
- Asynchronous exam: 25-minute examination available for 3 hours, from 11:00 to 14:00.
- The questions are randomly selected from a pool of 100 questions.

They are categorised according to their topic and level of difficulty and, for each category, there are 10 questions.

Regarding examination participation, the percentage of enrolled students who took the examination in the current 2019-20 semester was significantly higher than in the three previous semesters, for which the percentage of students who took the examination was between 62% and 67%. This rate reached 79% in the 2019-20 academic semester. A total of 105 students took the examination, out of whom 81 (77%) passed. Table I shows the examination grades in the previous four semesters. The lecturers teaching in the four semesters have been the same and, the examinations designed were of the same difficulty. Comparing the 2019-20 academic semester with the previous semesters, the grades are significantly better. For example, 59%, 71% and 45% of students failed in this examination in previous semesters with a face-to-face examination, while only 23% of students who completed the examination failed in the 2019-20 semester.

TABLE I. GRADES FOR THE EXAMINATION

Grades	2016-17	2017-18	2018-19	2019-20
A	0%	0%	3%	11%
B	9%	2%	12%	40%
C	32%	27%	40%	26%
D	59%	71%	45%	23%

#### B. Assignments

Throughout the 2019-20 semester, the students had to submit five assignments (A1 to A5) to the platform within a defined deadline. The assignments had the following characteristics:

- There are six lab groups, in each of which an assignment was proposed. Therefore, the assignment that each student had to submit depended on the practice group he/she was assigned.
- The students had four days to complete and submit each assignment.
- Students must perform the assignment individually.

Lecturers know that students have programmed in groups in previous programming subjects and, in many cases, they are used to working in this way. Therefore, it is likely that two students who work together beforehand will continue to work together in this subject and probably also submit the assignment simultaneously. However, the lecturers encouraged solving the assignments individually because the final assessment is individual.

Table II shows the total number of students who submitted each assignment and the total number of students who did not.

TABLE II. TOTAL NUMBER OF ASSIGNMENTS SUBMITTED AND NOT SUBMITTED DURING THE 2019-20 ACADEMIC SEMESTER

Assignments	A1	A2	A3	A4	A5
Submitted	112	100	109	102	87
Not submitted	20	32	23	30	45

We consider the assignments to analyse whether there is a correlation between the clusters of students who can be detected taking the examination together and those clusters of students who worked together on assignment submissions.

### V. RESULTS

Once all of the students had finished the examination, the lecturer downloaded the examination records. Surprisingly, the students' completion of the examination was evenly distributed over the

<sup>1</sup> <https://github.com/abalderas/Py-Cheat>

3-hour interval that it was available (see Fig. 4). This distribution was unexpected because the students did not have any other overlapping classes. In the following subsections, we analyse the examination records through the Py-Cheat tool to find evidence of the students' fraudulent behaviour. Next, we analyse the records of the assignments carried out by the students throughout the course. In this analysis, we look for evidence of collaboration similar to those found in the examination.

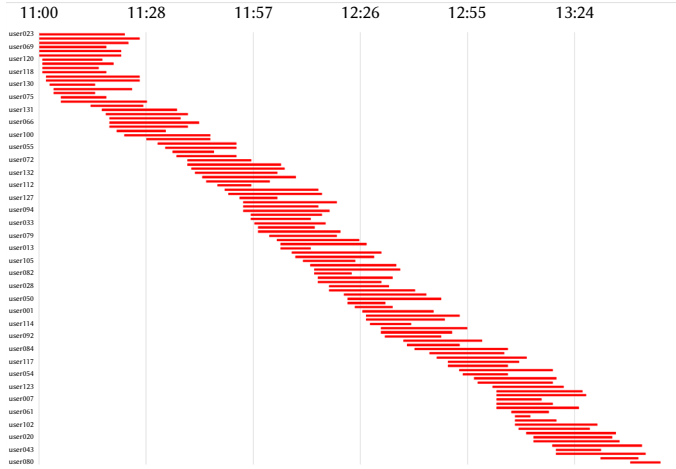


Fig. 4. Time distribution of the examination between the 3-hours slot.

### A. Exam Analysis

We first draw clusters of two or more students who have sequentially taken the examination with a time interval of up to two minutes between the first student ( $S_1$ ) submitting the examination and the second student ( $S_2$ ) starting it; that is,  $ST_2 - FT_1 \leq 2min$ . The results obtained indicate that 71 students met this pattern (see Fig. 5). Students assigned to the same cluster is represented by the colour of the nodes.

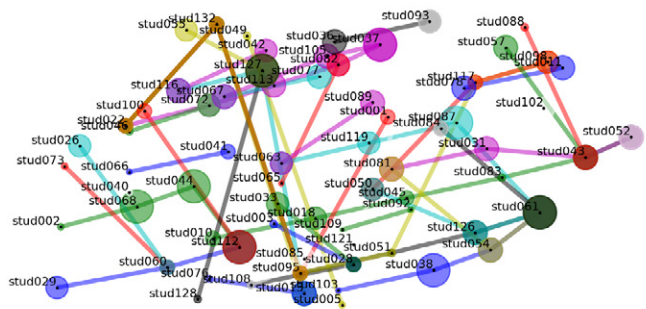


Fig. 5. Clusters of two or more students taking their examination consecutively within a 2-minute interval.

In Fig. 6, we prune the previous network by keeping only the clusters of at least five students who have taken the test consecutively with an interval of fewer than 2 minutes. In this case, four clusters of students appear. It is worth noting that some clusters overlap. Stud022 is the first to start the examination. When they finishes, stud132 and stud072 start. When stud051 finishes, stud117 and stud126 start, which would imply that there are clusters of students who worked in parallel. The diameter of the nodes represents the student's grade. The nodes have a larger diameter when they are closer to the deadline (i.e., the later they take the examination).

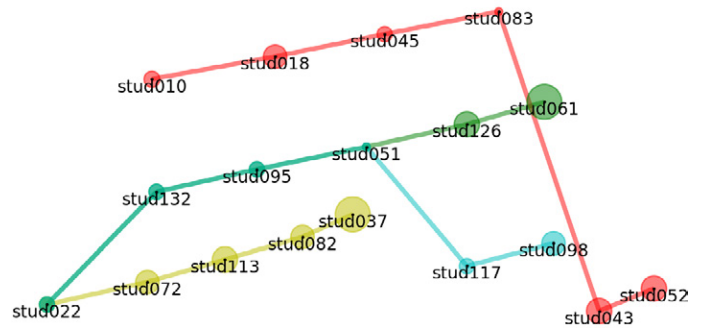


Fig. 6. Clusters of five or more students taking their examination consecutively within a 2-minute interval.

Finally, Fig. 7 shows clusters of six or more students who consecutively took the examination within a 5-minute interval. Table III shows the data for one cluster of students who took the examination with a time interval of fewer than 5 minutes. In this case, the first student took 25 minutes to take their examination and, from that point on, the time decreases until the sixth student takes only 10 minutes. Grades tend to be better for the last students to take the examination. Although the multiple-choice questions are randomly selected from a pool of 100 questions, there may be repetitions as new examinations are generated. Assuming that a cluster of students collaborate to take the examination, the last few students who take the examination would benefit from the repeated questions. It is worth noting that in this cluster of students, five belonged to the same laboratory group (G2).

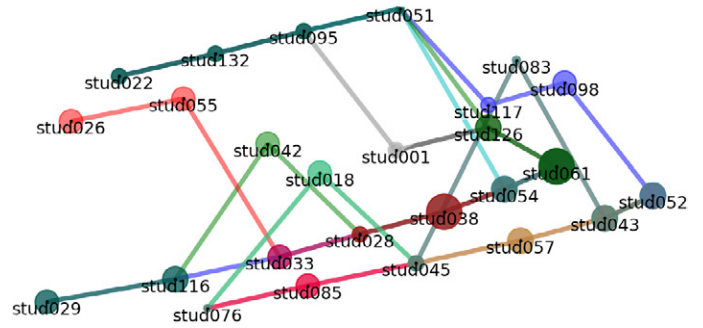


Fig. 7. Clusters of six or more students taking their examination consecutively within a 5-minute interval.

TABLE III. CLUSTER OF STUDENTS SEQUENTIALLY TAKING THE EXAMINATION

Student	Lab	Start	Finish	Time	Grade	Grade/min
stud076	G3	11:40	12:05	25	3.75	0.15
stud018	G2	12:08	12:32	24	7.50	0.31
stud045	G2	12:32	12:55	23	6.25	0.27
stud057	G2	12:58	13:18	20	8.75	0.43
stud043	G2	13:19	13:31	12	8.75	0.73
stud052	G2	13:31	13:41	10	8.75	0.87

### B. Assignment Analysis

Concerning assignments, we considered two possible scenarios. First, given that it is very typical to work in pairs in many courses of the degree, we checked clusters of two or more students who would submit assignments within an interval of fewer than 5 minutes. Based on the evidence collected, 97 students collaborated on some of the assignments with at least one classmate.

Figure 8 shows the clusters of students who worked together on assignment A3. Most of the clusters submitted their assignment when it had just been activated (left cluster) and when the deadline was close (right cluster). Meanwhile, only two pairs of students performed their work at an intermediate point (middle cluster).

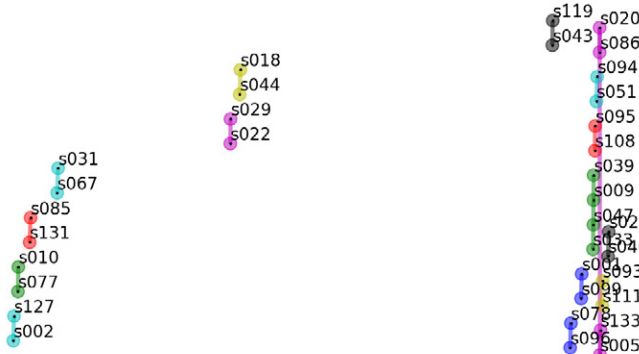


Fig. 8. Clusters of two or more students submitting their assignment consecutively within a 5-minute interval for assignment A3.

The clusters of students detected in the 5-minute interval for assignments are frequently composed of two students (Table IV). Students complete the assignment in collaboration: one student submits it first and another does it afterwards. For assignments A2, A3 and A4, Py-Cheat identifies 12, 13 and 10 pairs of students, respectively. The largest cluster of students detected was of eight students for assignment A1.

TABLE IV. TOTAL NUMBER OF CLUSTERS DETECTED FOR EACH ASSIGNMENT WITH THE NUMBER OF STUDENTS INDICATED IN THE COLUMN HEADER: 5-MINUTES INTERVAL, MINIMUM OF 2 STUDENTS

Student per cluster	2	3	4	5	6	7	8
A1	4	6	0	1	1	2	1
A2	12	1	1	0	0	0	0
A3	13	0	2	0	0	0	0
A4	10	0	0	0	0	0	0
A5	4	2	0	0	0	0	0

After comparing these data with those obtained in the examination (clusters of two or more students taking their examination consecutively within a 5-minute interval), we detected that 46 students appear in collaborative clusters in both activities. In other words, we found evidence that 46 students first collaborated in the assignments and then collaborated in the multiple-choice questionnaire.

Second, the lecturer looked for large clusters of students (five or more) who coordinated to complete the assignments with an interval of fewer than 10 minutes between one submission and the next. Following this approach, 51 students participated in clusters of at least five students in the five assignments of the course (see Fig. 9).

As summarised in Table V, Py-Cheat detected large clusters of students for the first three assignments (A1, A2 and A3). Two clusters of 10 students stand out for assignment A1. However, for the last assignments of the course (A4 and A5), Py-Cheat did not detect large clusters of students.

If we compare these data with the clusters of two or more students taking their examination consecutively within a 5-minute interval, we found evidence that 24 students first collaborated in the assignment and then collaborated in the multiple-choice questionnaire.

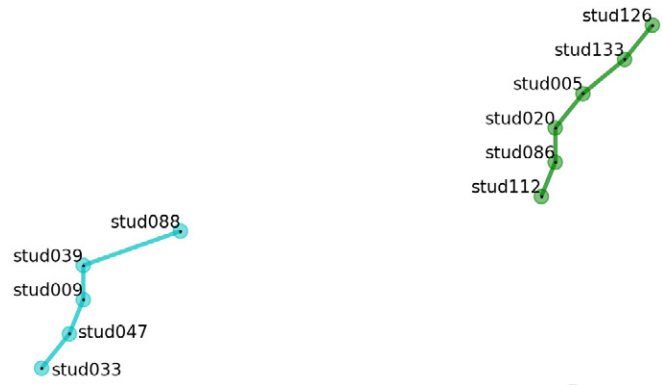


Fig. 9. Clusters of five or more students submitting their assignment consecutively within a 10-minute interval for assignment A3.

TABLE V. TOTAL NUMBER OF CLUSTERS DETECTED FOR EACH ASSIGNMENT WITH THE NUMBER OF STUDENTS INDICATED IN THE COLUMN HEADER: 10-MINUTES INTERVAL, MINIMUM OF FIVE STUDENTS

Student per cluster	5	6	7	8	9	10
A1	0	1	1	1	0	2
A2	1	0	0	0	0	0
A3	1	1	0	0	0	0
A4	0	0	0	0	0	0
A5	0	0	0	0	0	0

### C. Validation

To evaluate whether the collaboration detected through the Py-Cheat application has been significant, we compare the performance of students who successfully passed the examination with students from previous years.

After the questionnaire that we used in this study, the students were required to take an examination on SQL queries. In the following, we compare the dependence relationship among the three previous courses between the students who passed this questionnaire and those who passed the SQL queries examination. We will then perform the same comparison with the students in the case study (Table VI).

TABLE VI. RESULTS OF PREVIOUS FOUR ACADEMIC SEMESTERS

Semester	Stud	Activity	Passes	Fails
2016-17	115	Questionnaire	27.83%	72.17%
		SQL queries exam	26.09%	73.91%
2017-18	89	Questionnaire	20.22%	79.78%
		SQL queries exam.	29.21%	70.70%
2018-19	80	Questionnaire	51.25%	48.75%
		SQL queries exam.	33.75%	66.25%
2019-20	105	Questionnaire	66.67%	33.33%
		SQL queries exam.	34.29%	65.71%

To carry out this comparison, we define the following null hypothesis:

- $H_0$ : Passing the questionnaire and the SQL queries examination is not related. There is no dependency between students who have passed the SQL queries examination and those who previously passed the questionnaire.

Therefore, the alternative hypothesis would be stated as follows:

- $H_1$ : Passing the questionnaire and the SQL queries examination is related; that is, students who passed the SQL queries examination

may have previously passed the questionnaire.

To determine the dependency between passing the SQL queries examination after passing the questionnaire, we used the Chi-square test with a significance level of 0.05. For the 2016-17, 2017-18 and 2018-19 academic semesters, we obtain a p-value of 13.1490, 20.1829 and 23.1090, respectively. All three values are above the significance threshold of 0.05 ( $X^2 > 3.8414$ ), so for the previous three courses in which the examination was face-to-face, we cannot accept the null hypothesis. Thus, we assume that there is a relationship between first passing the questionnaire and then passing the SQL queries examination.

However, for the 2019-20 academic semester, we obtained a p-value of 0.7608 in the Chi-square test. This value is lower than that established for a significance value of 0.05 ( $X^2 < 3.8414$ ). Thus, the null hypothesis cannot be discarded and this supports the fact that some students have cheated. This under-performance between this cohort of students and that of previous semesters concerning passing the second examination after having passed the first supports the view that students in the online examination cheated more than in face-to-face evaluation.

## VI. DISCUSSION

The results show evidence of cheating behaviour by students based on fraudulent collaboration among them, both for course assignments and during the online questionnaire.

Why do students cheat? As Albrecht states [9], students cheat when they know that they will not be caught cheating. The lecturer cannot see what the students are doing at home while they are taking the examination. Furthermore, they are motivated to cheat by the pressure of an official examination.

Finally, Albrecht pointed to a third reason, which is the rationalisation of the action. With Py-Cheat, evidence was obtained indicating that up to 97 students may have fraudulently collaborated in their class assignments. Therefore, if the students are familiar with collaboration during in-class assignments, then they can normalise collaborating in an examination—even if it goes against the rules. The environment is the same (online), from home they can still communicate with their classmates via videoconference, phone call or messaging while the lecturer is unaware of their actions. Consequently, they rationalise their behaviour as doing the same routine that they used for assignments.

For the assignments, the collaborations detected were mainly from pairs of students. Depending on the setting used, i.e. interval between submission of assignments and the minimum number of students per cluster, evidence has been found that between 24 and 46 students collaborated first on the assignments and then on the questionnaire. This finding is in line with the research of Hellas et al. [17]. They found that computer engineering students who had practised pair programming on assignments worked also together on the take-home examination. This work focused on the search for plagiarism based on the similarity of responses. This approach is similar to other tools such as Turnitin or Viper [18], but it also allows for detecting copy-paste patterns. To carry out this, the student had to install a plugin.

The method we present allows detecting evidence of cheating regardless of the type of activity, as it considers the time of submission, no matter if they are multiple-answer questions or a piece of code. Besides, it is transparent to the student, as no plugin is required.

The use of submission timestamps to detect cheating is in line with Tiong and Lee's research [31]. They use the submission timestamp to prevent students from cheating. Compared to our method, they aim at cheating prevention instead of detection. In this paper, different

multiple-choice examinations were defined with a set of questions each. If the student answered a question in less time than expected, the system changed the following questions in his/her exam. Although the authors indicated that this was an effective system to prevent cheating, other work has shown that personalising examinations can be unfair to students [30].

Our method for detecting cheating does not depend on how the lecturer sets up the questionnaire. It considers the time of submission and the grades obtained to provide evidence of fraudulent collaboration. Therefore, it does not generate unfair situations per se, as it is transparent to the student and merely reflects what is recorded on the virtual campus.

Concerning the questionnaire, larger clusters of students took the questionnaire sequentially. While one student in the cluster takes the examination (examinee), the other students in the cluster help the former to solve the exam (collaborators). In this way, they exchange their roles and perform the examination sequentially until the last one finishes. Generally, students obtain a higher grade when their position in the sequence is closer to the end. In addition, the first students in a sequence typically take longer to complete the test than the last students. As an example, we can look at Table III, where the first student in a sequence of six took 25 minutes, while the sixth student took only 10 minutes.

These results are similar to those detected in the research of Jaramillo et al. [32]. They used submission timestamps and the responses selected by students to detect their cheating behaviour. However, their method relies on questions being repeated between exams. This is contrary to good practice in online exam design, such as using question randomisation and large question pools. The evidence based on the submission timestamps collected in our work is independent of the answers submitted by the students.

This study was made possible thanks to Py-Cheat because the tool automated the extraction of evidence of student collaboration in activities and examinations conducted through an LMS. In this study, the evidence was used to confirm suspicions of fraudulent students' behaviour. Our method avoids privacy issues caused by tools such as Proctoring or Respondus. These tools use the webcam to monitor how students take the examination and, although they can be effective to detect cheaters, this causes controversy within the educational community [22].

The aim of this paper is not to investigate the psychology of students and why they cheat. Even more, based on the results gathered in this paper, we do believe that lecturers should be encouraged to use an assessment based on continuous evaluation of their courses instead of final examinations [33], [34]. Additionally, thanks to learning analytics, we can collect learning records that can be used for evidence-based assessment [35], [36].

At the end of the course, we invited students to anonymously answer a survey in which they responded to different aspects of online teaching. We asked them what caused them the main difficulty in following the course, and most of their answers mentioned problems stemming from the situation generated by the pandemic: family problems, stress, anxiety or difficulty in concentrating.

This case study took place during the Covid-19 pandemic, which has affected the whole society, including students [37]. However, the grades achieved by the students were surprisingly higher than in previous years. For this improvement, we can find two justifications. On the one hand, the students did improve over previous years [38]. However, this would be unexpected, considering the low attendance and participation in class. On the other hand, it would be more possible for the pandemic to push some students to engage in or normalise this fraudulent behaviour during examinations.

## VII. CONCLUSIONS

The Covid-19 pandemic significantly affected Spanish higher education in 2020 spring semester, forcing teaching and assessment to shift from face-to-face to online overnight. Among other issues, the assessment activities had to be conducted online, which made fraudulent activities harder to detect. This research presents Py-Cheat, a tool to detect students' fraudulent collaboration in the submission of assignments and examinations through their LMS activity records. Specifically, the method is based on indicators as the submission timestamps and the grades obtained. The evaluation case study shows evidence that a large number of students cheated during an examination based on a multiple-choice questionnaire. The students were organised in clusters and sequentially took their examinations in collaboration.

The results collected are promising. In a virtual context, where the lecturer cannot know what the students are doing, Py-Cheat provides evidence concerning the students' behaviour and it graphically draws the different clusters of students who collaborate in the completion of assignments.

We recommend the implementation of plans to raise awareness of the ethics code and dissuade cheating. We also encourage the use of learning analytics techniques and tools such as Py-Cheat to detect fraudulent behaviour among students in the performance of assignments and examinations in the context of normality.

In our future work, this tool will focus on detecting patterns of students' collaboration and incorporating new evidence to assess whether these collaborations can lead to an improved student performance.

## ACKNOWLEDGMENT

This research was funded by Spanish National Research Agency (AEI), through the project VISAIGLE (TIN2017-85797-R) with ERDF funds.

## REFERENCES

- [1] D. Stuber-McEwen, P. Wiseley, S. Hoggatt, "Point, click, and cheat: Frequency and type of academic dishonesty in the virtual classroom," *Online Journal of Distance Learning Administration*, vol. 12, no. 3, 2009.
- [2] F. J. García-Peñalvo, "El sistema universitario ante la covid-19: Corto, medio y largo plazo," *Universidad*, 2020, doi: <https://bit.ly/2YPUeXU>.
- [3] C. Giovannella, M. Passarelli, D. Persico, "The effects of the covid-19 pandemic on italian learning ecosystems: The school teachers' perspective at the steady state," *Interaction Design and Architecture(s)*, vol. 45, pp. 264–286, 2020.
- [4] R. Harper, T. Bretag, K. Rundle, "Detecting contract cheating: examining the role of assessment type," *Higher Education Research & Development*, vol. 40, no. 2, pp. 263–278, 2021.
- [5] B. Chen, M. West, C. Zilles, "How much randomization is needed to deter collaborative cheating on asynchronous exams?," in *Proceedings of the Fifth Annual ACM Conference on Learning at Scale*, 2018, pp. 1–10.
- [6] I. M. Sierra, M. G. Gómez, J. S. Eizaguirre, "Learning analytics for formative assessment in engineering education," *The International Journal of Engineering Education*, vol. 34, no. 3, pp. 953–967, 2018.
- [7] A. Álvarez-Arana, M. Larrañaga-Olagaray, M. Villamañe-Gironés, "Mejora de los procesos de evaluación mediante analítica visual del aprendizaje," *Education in the Knowledge Society*, vol. 21, no. 9, 2020, doi: 10.14201/eks.21554.
- [8] R. A. Bernardi, R. L. Metzger, R. G. S. Bruno, M. A. W. Hoogkamp, L. E. Reyes, G. H. Barnaby, "Examining the decision process of students' cheating behavior: An empirical study," *Journal of Business Ethics*, vol. 50, no. 4, pp. 397–414, 2004.
- [9] W. S. Albrecht, G. W. Wernz, T. L. Williams, et al., *Fraud: Bringing light to the dark side of business*. Irwin Professional Pub., 1995.
- [10] E. J. Austin, D. H. Saklofske, S. M. Mastoras, "Emotional intelligence, coping and exam-related stress in canadian undergraduate students," *Australian Journal of Psychology*, vol. 62, no. 1, pp. 42–50, 2010.
- [11] E. T. Baloran, "Knowledge, attitudes, anxiety, and coping strategies of students during covid-19 pandemic," *Journal of Loss and Trauma*, vol. 25, no. 8, pp. 635–642, 2020.
- [12] F. J. García-Peñalvo, A. Corell, V. Abella-García, M. Grande-de Prado, "Recommendations for mandatory online assessment in higher education during the covid-19 pandemic," in *Radical Solutions for Education in a Crisis Context*, Springer, 2021, pp. 85–98.
- [13] G. R. Watson, J. Sottile, "Cheating in the digital age: Do students cheat more in online courses?," *Online Journal of Distance Learning Administration*, no. 13.1, 2010.
- [14] S. Kocdar, A. Karadeniz, R. Peytcheva-Forsyth, V. Stoeva, "Cheating and plagiarism in e-assessment: students' perspectives," *Open Praxis*, vol. 10, no. 3, pp. 221–235, 2018.
- [15] R. W. Smith, T. Prometric, "The impact of braindump sites on item exposure and item parameter drift," in *Annual Meeting of the American Education Research Association*, 2004.
- [16] T. Mason, A. Gavrilovska, D. A. Joyner, "Collaboration versus cheating: Reducing code plagiarism in an online ms computer science program," in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 2019, pp. 1004–1010.
- [17] A. Hellas, J. Leinonen, P. Ihanntola, "Plagiarism in take-home exams: Help-seeking, collaboration, and systematic cheating," in *Proceedings of the 2017 ACM conference on innovation and technology in computer science education*, 2017, pp. 238–243.
- [18] R. R. Naik, M. B. Landge, C. N. Mahender, "A review on plagiarism detection tools," *International Journal of Computer Applications*, vol. 125, no. 11, 2015.
- [19] A. Abdi, N. Idris, R. M. Alguliyev, R. M. Aliguliyev, "Pdtk: Plagiarism detection using linguistic knowledge," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8936–8946, 2015.
- [20] S. E. Eaton, K. L. Turner, "Exploring academic integrity and mental health during covid-19: Rapid review," *Journal of Contemporary Education Theory & Research (JCETR)*, vol. 4, no. 2, pp. 35–41, 2020.
- [21] B. Chen, S. Azad, M. Fowler, M. West, C. Zilles, "Learning to cheat: Quantifying changes in score advantage of unproctored assessments over time," in *Proceedings of the Seventh ACM Conference on Learning@Scale*, 2020, pp. 197–206.
- [22] C. S. Gonzalez-Gonzalez, A. Infante-Moro, J. C. Infante-Moro, "Implementation of e-proctoring in online teaching: A study about motivational factors," *Sustainability*, vol. 12, no. 8, p. 3488, 2020.
- [23] A. Balderas, J. M. Doderó, M. Palomo-Duarte, I. Ruiz-Rube, "A domain specific language for online learning competence assessments," *International Journal of Engineering Education*, vol. 31, no. 3, pp. 851–862, 2015.
- [24] M. L. Sein-Echaluce, A. Fidalgo-Blanco, J. Esteban-Escano, F. J. García-Peñalvo, M. A. Conde-González, "Using learning analytics to detect authentic leadership characteristics in engineering students," *International Journal of Engineering Education*, vol. 34, no. 3, pp. 851–864, 2018.
- [25] A. Balderas, L. De-La-Fuente-Valentin, M. Ortega-Gomez, J. M. Doderó, D. Burgos, "Learning management systems activity records for students' assessment of generic skills," *IEEE access*, vol. 6, pp. 15958–15968, 2018.
- [26] A. Balderas, M. Palomo-Duarte, J. M. Doderó, M. S. Ibarra-Sáiz, G. Rodríguez-Gómez, "Scalable authentic assessment of collaborative work assignments in wikis," *International Journal of Educational Technology in Higher Education*, vol. 15, no. 1, pp. 1–21, 2018.
- [27] F. Riquelme, R. Munoz, R. Mac Lean, R. Villarroel, T. S. Barcelos, V. H. C. de Albuquerque, "Using multimodal learning analytics to study collaboration on discussion groups," *Universal Access in the Information Society*, vol. 18, no. 3, pp. 633–643, 2019.
- [28] D. Von Gruenigen, F. B. d. A. e Souza, B. Pradarelli, A. Magid, M. Cieliebak, "Best practices in e-assessments with a special focus on cheating prevention," in *2018 IEEE Global Engineering Education Conference (EDUCON)*, 2018, pp. 893–899, IEEE.
- [29] S. Manoharan, "Cheating-resistant multiple-choice examinations using personalization," *Computers & Education*, vol. 130, pp. 139–151, 2019.
- [30] P. Denny, S. Manoharan, U. Speidel, G. Russello, A. Chang, "On the



fairness of multiple-variant multiple-choice examinations,” in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 2019, pp. 462–468.

- [31] L. C. O. Tiong, H. J. Lee, “E-cheating prevention measures: Detection of cheating at online examinations using deep learning approach—a case study,” *arXiv preprint arXiv:2101.09841*, 2021.
- [32] D. Jaramillo-Morillo, J. Ruipérez-Valiente, M. F. Sarasty, G. Ramírez-Gonzalez, “Identifying and characterizing students suspected of academic dishonesty in spocs for credit through learning analytics,” *International Journal of Educational Technology in Higher Education*, vol. 17, no. 1, pp. 1–18, 2020.
- [33] K. A. Villanueva, S. A. Brown, N. P. Pitterson, D. S. Hurwitz, A. Sitomer, “Teaching evaluation practices in engineering programs: Current approaches and usefulness,” *International Journal of Engineering Education*, vol. 33, no. 4, pp. 1317–1334, 2017.
- [34] F. J. García-Peñalvo, A. Corell, V. Abella-García, M. Grande, “Online assessment in higher education in the time of covid-19,” *Education in the Knowledge Society*, vol. 21, 2020.
- [35] Á. Fidalgo-Blanco, M. L. Sein-Echaluze, F. J. García-Peñalvo, M. Á. Conde, “Using learning analytics to improve team-work assessment,” *Computers in Human Behavior*, vol. 47, pp. 149–156, 2015.
- [36] M. Palomo-Duarte, A. Berns, A. Balderas, J. M. Dodero, D. Camacho, “Evidence-based assessment of student performance in virtual worlds,” *Sustainability*, vol. 13, no. 1, p. 244, 2021.
- [37] Chegg.org, “Global student survey 2021,” Chegg Inc., 2021. [Online]. Available: <https://www.chegg.org/global-student-survey-2021>.
- [38] S. Iglesias-Pradas, Á. Hernández-García, J. Chaparro-Peláez, L. Prieto, “Emergency remote teaching and students’ academic performance in higher education during the covid-19 pandemic: A case study,” *Computers in Human Behavior*, p. 106713, 2021.



Antonio Balderas

Antonio Balderas received his MSc degree in computer science and his PhD degree from the University of Cadiz, Spain. He is currently with the University of Cadiz, and works as an Assistant Professor in the Department of Computer Engineering and as a Researcher with the Software Process Improvement and Formal Methods Group. He was a project manager in different Spanish IT

companies. His research interests include technology-enhanced learning and creative computing.



Manuel Palomo-Duarte

Manuel Palomo-Duarte received his MSc degree in computer science from the University of Seville and his PhD degree from the University of Cadiz. He works in the Computer Science Department of the University of Cadiz as an Associate Professor. He is the author of more than 20 papers published in indexed journals and than 30 contributions to international academic conferences.

His main research interests are learning technologies, serious games and collaborative web. He was a board member of Wikimedia Spain from 2012 to 2016.



Juan Antonio Caballero-Hernández

J. A. Caballero-Hernández received his MSc degree in computer science and his PhD degree from the University of Cadiz, Spain. His main research interest is focused on learning experiences based on serious games and diverse applications of process mining. Outside the academic environment, he has worked in many different positions in the IT sector, including web development, managing teams and international projects.



Mercedes Rodriguez-Garcia

Mercedes Rodriguez-Garcia is an assistant lecturer in the Department of Automation Engineering, Electronics and Computer Architecture at the University of Cádiz, Spain. She received her Ph.D. in Computer Science and Mathematics of Security from the Universitat Rovira i Virgili in 2017. Her research interests include data privacy, computer network security, and reverse engineering and secure architectures.



Juan Manuel Dodero

Juan Manuel Dodero received his degree in CS from the Polytechnic University of Madrid and his PhD degree in CS from the Carlos III University of Madrid. He was a Research and Development Engineer with Intelligent Software Components S.A. He has been a lecturer with the Carlos III University of Madrid. He is currently a Full Professor with the University of Cadiz, Spain. His

main research interests include Web science, and engineering and technology-enhanced learning; fields in which he has co-authored numerous contributions in journals and research conferences.