Supporting Skill Assessment in Learning Experiences Based on Serious Games Through Process Mining Techniques

Juan Antonio Caballero-Hernández[1]*, Manuel Palomo-Duarte[2], Juan Manuel Dodero[3], Dragan Gaševic[4]

[1] EVALfor research group, University of Cadiz, Puerto Real (Spain)
[2] Department of Computer Science, University of Cadiz, Puerto Real (Spain)
[3] Department of Computer Science, University of Cadiz, Puerto Real (Spain)
[4] Faculty of Information Technology, Monash University, Melbourne (Australia)

* Corresponding author. juanantonio.caballero@uca.es

Received 26 September 2021 | Accepted 11 February 2023 | Early Access 9 May 2023

Abstract

Learning experiences based on serious games are employed in multiple contexts. Players carry out multiple interactions during the gameplay to solve the different challenges faced. Those interactions can be registered in logs as large data sets providing the assessment process with objective information about the skills employed. Most assessment methods in learning experiences based on serious games rely on manual approaches, which do not scalewell when the amount of data increases. We propose an automated method to analyse students' interactions and assess their skills in learning experiences based on serious games. The method takes into account not only the final model obtained by the student, but also the process followed to obtain it, extracted from game logs. The assessment method groups students according to their in-game errors and in-game outcomes. Then, the models for the most and the least successful students are discovered using process mining techniques. Similarities in their behaviour are analysed through conformance checking techniques to compare all the students with the most successful ones. Finally, the similarities found are quantified to build a classification of the students' assessments. We have employed this method with Computer Science students playing a serious game to solve design problems in a course on databases. The findings show that process mining techniques can palliate the limitations of skill assessment methods in game-based learning experiences.

Keywords

Educational Process Mining, Game-Based Learning, Learning Analytics, Model Discovery, Serious Games, Skill Assessment.

DOI: 10.9781/ijimai.2023.05.002

I. Introduction

Serious games are considered to be those games which have purposes beyond entertainment [1]. The employment of serious games in educational contexts is promising to create and develop learning processes where students are actively involved. A lot of research on the positive impact and outcomes associated with playing serious games can be found in the literature [2]. Although serious games are widely employed in online learning, the methods for their assessment still rely on manual approaches, which are scarce in details of the assessment of the learning outcomes, have scalability problems, and lack automated and semi-automated support [3]. During online gameplay, players can carry out diverse interactions according to the features of the game, e.g.: movements, such as jumping or running, selections of a text option in a conversation, selection of an item using the pointer, etc. These interactions can remain in storage records, databases, or log files, thus resulting in data sets that include objective information about the skills employed during the game. The analysis of such large data sets can lead to scalability problems when using manual methods of assessment.

The processes of Learning Analytics (LA) can palliate these limitations through data-driven analysis. LA is "the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs" [4]. The interactive nature of serious games makes them a significant source of data, tracking user interactions and storing them as sequential events in a log. Among the sequence analysis techniques, process mining can be used to discover, monitor and improve the actual processes by extracting knowledge from an event log [5]. As a discipline, process mining is situated between computational intelligence, data mining, and process modeling and analysis. Due to the sequential nature of

Please cite this article as: J. A. Caballero-Hernández, M. Palomo-Duarte, J. M. Dodero, D. Gaševic, "Supporting Skill Assessment in Learning Experiences Based on Serious Games Through Process Mining Techniques", International Journal of Interactive Multimedia and Artificial Intelligence, vol. 8, no. 6, pp. 146-159, 2024, http://dx.doi.org/10.9781/ijimai.2023.05.002

game interactions and the identified limitations of their assessment, the techniques for their event-based data analysis are considered to provide the basis of an automated method to support skill assessment in learning experiences based on serious games.

This paper proposes a method based on a combination of process mining techniques to analyse the logs produced by a serious game. We have developed process models and carry out a conformance check to validate them with particular event logs. Our method conducts a performance comparison using assessment models or profiles, inspired by existing assessment integration approaches [6]. Assessment models can be considered as good examples of the behaviour that can be imitated by the students. The proposed method provides an assessment model as well as indicators to measure the gap between the behaviour represented by the assessment model and the behaviour of the students. The method conducts a more detailed assessment through behaviour analysis to identify similarities and differences between each student and those students who are most successful in their learning outcomes. The method considers objective evidence from the process carried out by the student during the experience, extracted from the game logs.

In order to validate our proposal, a case study was carried out in the context of a Computer Science degree program. More than 100 students enrolled in a course on databases played a serious game and designed a conceptual data specification through an Entity–Relationship (E/R) diagram. In playing the game, the students were expected to use a specific skill for the analysis and design of relational databases, focused on the learning outcome related to the E/R diagram modelling, namely "knowledge to produce a logical and conceptual design of a database."

The rest of this paper is organized as follows. In Section II we place the subject of serious games in its context, as well as learning assessment methods, as found in the literature. Section III introduces process mining and defines its main concepts. Then, the research questions and the proposal are described in Section IV. The conducted case study is detailed in Section V. A discussion of the results of implementing our process mining proposal is presented in Section VI. Finally, we list the conclusions of the study and identify future lines of work in Section VII.

II. Learning Assessment in Serious Games

Assessment in the educational process validates the acquisition of skills by the students. Allen defines assessment as "the use of empirical data on student learning to refine programs and improve student learning" [7]. Despite the variety of assessment methods found in the literature, most of their implementations rely on manual assessment [3]. Manual assessment covers all those processes supported by traditional approaches to assessing student skills, such as instructor observations, which might be based on subjective assumptions, and traditional tests, where the answers to each question are seen as independent data points. However, learning and succeeding in a complex and dynamic world is not easily measured by multiple-choice responses on a simple knowledge test [8]. Manual assessment can suffer from problems in assessing large data sets, due to scalability limitations and the lack of automated and semi-automated mechanisms to support the assessment. In addition, serious games are commonly employed with formative aims, while the assessment of the acquired skills is implemented through external tools with predefined answers, resulting in possible omissions of relevant information.

Some well-known areas related to LA are usually employed to assessment in serious games to palliate some of these limitations. Evidence-centred assessment design (ECD) is a framework to provide language, concepts and knowledge representations for designing and delivering learning assessments, organized around the evidentiary argument an assessment is meant to embody [9]. ECD contains a conceptual assessment framework (CAF) layer, widely used for educational assessment development and considered the blueprint for an assessment [10]. CAF is divided into models, where each model provides the answer for critical questions related to the assessment process. Student models define one or more variables related to the skills we wish to measure. Evidence models provide concrete instructions for analysing and measuring the variables defined within the student model. Task models describe the situations in which to obtain the evidence needed for the evidence models.

Following the ECD approach, stealth assessment is an embedded and in-process assessment, usually focused on formative aims. Stealth assessment aims to support learning and keep the student engaged in the activity while removing or reducing test anxiety without sacrificing reliability and validity [8]. Stealth assessment is seamlessly included in the educational process. It represents a quiet process by which student's interactions involving the levels of the relevant skills are stored in a dynamic model [11].

One widely used tool for stealth assessment is a Bayesian network [8]. Bayesian networks can be used within student models to handle uncertainty by using probabilistic inference to update and improve belief values (i.e., regarding student skills). Bayesian networks have been used in assessment systems where player interactions are captured during the game and related key indicators provide evidence for the skills employed [12]. Bayesian networks have also been used to assess students' performance in intelligent tutoring systems. Three constraint-based intelligent tutoring systems focused on database education are presented by Mitrovic et al. [13]. That approach provides feedback to students according to a description of the basic principles and concepts in the domain. One of these intelligent tutoring systems, called KERMIT, focuses on database modeling. KERMIT has been evaluated to prove its effectiveness according to the students' results [14].

An important and broadly used technique to model processes is Bayesian Knowledge Tracing (BKT) [15]. BKT is based on hidden Markov models and has been extensively used to perform assessments in intelligent tutoring systems [16]. BKT assumes that the student skills are represented as a set of binary variables. Each variable represents a skill that can be mastered by the student or not.

On the one hand, data-centred approaches tend to be agnostic as to the process: data mining, statistics, and machine learning do not consider end-to-end process models. On the other hand, process science approaches are process-centric but usually focused on modeling instead of discovering knowledge from the event data. Process mining is a mixed approach, between model-based process analysis and data-centred analysis [17]. Process mining seeks to confront the event data (i.e. the gathered evidence) with process models (generated automatically or hand made). In addition, process mining provides a more comprehensive and detailed picture of the structure of events that occur during a learning process, instead of having to aggregate process data into the frequencies or probabilities of events [18]. We consider that the unique position of process mining makes it a powerful tool for exploiting the growing availability of data from serious games and analyse in detail the processes performed by the students to be considered in the assessment.

III. Process Mining in Educational Environments

Process mining is considered a link between data mining and business process modeling and analysis [5]. An event log is the initial input in process mining. In general, an event log can be seen as a

collection of cases and a case can be seen as a trace of events. Any event must include some mandatory fields: a unique identificator per process instance called "CaseID", the corresponding "activity" and its "timestamp".

There are three types of process mining techniques: discovery, conformance and enhancement [17]. Discovery techniques use an event log produced by any type of process as input to produce a model without using any a priori information. Conformance checking techniques compare an existing model to an event log of the same process. Enhancement techniques are focused on improving or extending a process model through the information stored in an event log of the same process.

There are four desired criteria for the quality of a discovered model: fitness, precision, generalization and simplicity. Fitness measures how well the model is able to replay most of the traces in a log. Precision measures the model's acceptance of unrelated behaviour, so that a model with low precision is underfitting, thus enabling a completely different behaviour from the event log. Generalization measures how well the model can generalize the analysed behaviour: a model with a low generalization is too specific and too adapted to the behaviour of the event log. Lastly, simplicity uses Occam's Razor: "one should not increase, beyond what is necessary, the number of entities required to explain anything" [19]. Therefore, the discovered model should be the simplest model that represents the event log.

In those cases where the models are rather extreme, the scores for the quality criteria will be evident. However, it is more diÿcult to judge the quality of a model in realistic contexts. Conformance checking enables finding similarities and/or discrepancies between the modeled behaviour (discovered model) and the observed behaviour (event log) [17]. Conformance checking relates the events in the event log to the activities in the process model and compares them. The comparison is carried out by a "replay": a process to check whether each log trace can be simulated through the states of the discovered model. Using a replay, the fitness can be quantified to measure the similarities between the model and the event log.

Model discovery and conformance checking techniques enable the analysis of processes to provide behaviour models and compare their performance. However, correlating the different characteristics of a process can be essential to conducting a more refined analysis. These characteristics can be based on different perspectives, such as the control flow (i.e. the next interaction), the data flow (i.e. the age of the player), the time (i.e. the duration of the game) or the resource (i.e. the player who performed the action) [20]. In the assessment of students during a learning experience based on a serious game, it can be important to detect the errors made in the game and how they determined the final result of the game. Therefore, a correlation between the in-game errors and the results could determine the most decisive in-game errors.

Process mining has been widely used in multiple domains, including educational environments. Actually, an accepted term to refer to the use of process mining in educational environments is Educational Process Mining (EPM) [21]. EPM is focused on the use of the event logs registered by educational environments to discover, analyse and detect the most common behaviours performed during the learning process.

Different applications of EPM in higher education can be found in the literature. Model discovery has been used in multiple learning experiences, such as relating the students' performance to their studying behaviour or assessing wiki contributions during a collaborative experience [22], [23]. Conformance checking has also been used in studies focused on EPM, such as the introduction of the event-centred view of a process as a generally applicable approach for providing closer links between qualitative and quantitative research methods [24]. The work of Bannert and Reimann stands out in its use of EPM to identify process patterns in self-regulated learning [25],[26]. First, model discovery is employed to model the behaviour of different student profiles. A student profile can be considered as a group of students who share similar characteristics, such as outcomes or in-game errors. Comparing the students' profiles makes it possible to detect differences in their behaviour. Finally, a theoretical expert model is compared against the empirical trace data, using conformance checking.

IV. Process Mining Support of Skill Assessment in Serious Games

In this section, we present a method based on process mining techniques to support assessment in a learning experience based on serious games. First, the research question and sub-questions of this study will be presented. Then, the process followed to use the proposed process mining techniques and answer the research question and sub-questions will be given in detail.

A. Research Questions

Some limitations have been identified in the manual methods of assessing serious games: scalability problems, assessments that lack details, and a lack of automatic and semi-automatic mechanisms for the assessments. Considering these limitations and the possibilities provided by process mining, the main research question of our study is: *can process mining techniques support a scalable assessment in learning experiences based on serious games?* This question has been divided into the following sub-questions.

- RQ1: Can process mining techniques identify the most decisive in-game errors for specific student profiles?
- RQ2: Can process mining techniques detect similarities and differences between the performance of specific student profiles?
- RQ3: Can process mining techniques allow students to be assessed according to their performance during the game?

B. The Proposed Assessment Method

The proposed method addresses these research sub-questions by using different process mining techniques through the ProM open source framework, because it provides multiple functionalities for analysing large data sets and supports all the techniques included in our proposal [27]. The newest version (6.9, at the time this experiment was conducted) of ProM was used.

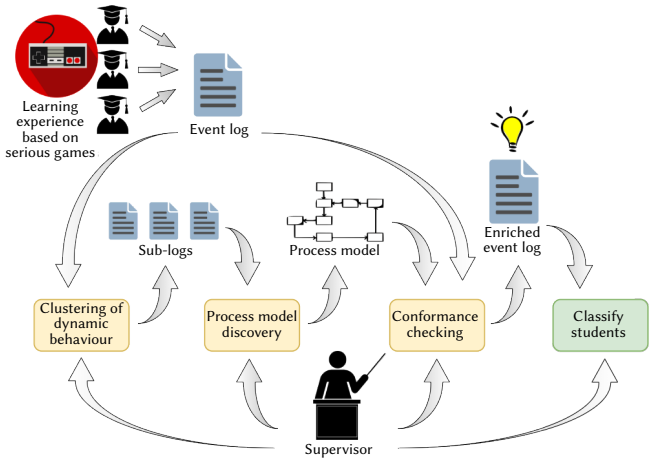

Fig. 1. Method to support the assessment of a learning experience based on serious games applying process mining techniques.

A case study has been conducted, using an event log produced by a serious game as the input for the proposed method, illustrated in Fig. 1. First, decision trees are generated to obtain trace clusters. This clustering makes it possible to detect different student profiles through a correlation analysis between students' outcomes and several types of in-game errors. Other existing approaches use different clustering techniques, such as agglomerative hierarchical clustering or k-means [28]. However, applying these techniques for trace clustering could not provide a clear insight into the characteristics of the process that are common to all the traces within the same cluster. Decision trees clearly highlight the discriminative characteristics, providing the reason why a certain log trace belongs to a given cluster and not to another [20].

Then, two subsets of the event log are obtained based on the previous clusters. These sub-logs contain detailed in-game errors for specific student profiles: the most successful students (MSS) and the least successful students (LSS). Behaviour similarities and differences between these student profiles are detected through model discovery techniques. The MSS model is used as an assessment model to be compared with each student's behaviour during gameplay through conformance checking. This comparison provides quantified data that represent the level of behaviour similarity between each student and the assessment model. This quantified data was added to the event log to enrich it. Finally, the enriched event log is explored to classify students according to their behaviour similarity in comparison to the assessment model. The stages of the proposed method are given in detail in the next subsections.

C. Clustering of Dynamic Behaviour

First, we aim to identify the most decisive in-game errors for the outcomes obtained by specific student profiles. This process is based on a successful system for clustering dynamic behaviour proposed and implemented in ProM [20]. Our starting point is an event log including all student gameplays in a learning experience based on serious games. During the game, the interactions performed by the students are stored in a log where each instance is a trace of events. These events must include some mandatory fields: the corresponding CaseID (a unique identificator per game experience of a student), the activity (performed interaction) and its timestamp. Depending on the context, additional fields can be included (i.e. grades achieved by a specific student, academic year, etc.). The in-game errors and final outcomes of each student were used during this case study, which is described in detail in Section V.

Starting from the event log, a process analysis can determine a so-called "analysis of a use case", which is defined as a triple $A = (c_r, C_d, F)$ consisting of [20]:

- Let C be the universe of characteristics,
  - c_r is a dependent characteristic defined as $c_r \in C \backslash C_d$,
  - C_d is a set of independent characteristics defined as $C_d \subseteq C \backslash \{c_r\}$,
- Let ε be the universe of events,
  - F is an event-selection filter defined as $F \subseteq \varepsilon$, which selects the events that are retained for the analysis.

Then, an analysis of a use case has to be defined to select a dependent characteristic, independent characteristics and an optional event filter. The behaviour analysis results in a decision tree whose purpose is to relate the dependent characteristic to the independent characteristics [20]. The ProM implementation of this system constructs decision trees relying on the algorithm C4.5 developed in the Weka toolkit [28].

Finally, the obtained decision tree can be used to cluster the executions of process instances with similar outcomes. In this tree, each trace is linked to one single instance that is associated with one leaf. All log traces associated with the same leaf are grouped within the same cluster, producing the same number of clusters as there are leaves in the tree. Event logs that include traces from multiple students are used to present a range of behavioural variability. Splitting the event log and grouping similar traces enables discovering partial models that are easier to understand and more representative than a discovered model produced by the whole event log. The employed implementation provides two well-known discretization techniques: equal-width binning and equal-frequency binning [29].

This process is structured as follows:

1. Input: Event log
2. Define an analysis of a use case ⟶ Use case
3. Analysis technique ⟶ Decision tree
4. Clustering decision tree ⟶ Sub-logs
5. if Sub-logs include [Generic behaviour]
6. Refine event log
7. Go to step 2
8. else //*Sub-logs include* [*Specific behaviour*]
9. Sub-logs include [Student profiles]

D. Discovery of a Process Model

We aim to detect the similarities and differences between the behaviours of specific student profiles. Based on the results of the clustering of the dynamic behaviour, clusters are selected to create sub-logs that represent these student profiles. Process patterns of the student profiles are analysed through discovery techniques: Inductive Miner (IM) and its variant – infrequent (IMi) [30].

First, IM aims to discover from any given event log a set of process models that fit the observed behaviour. Then, IMi adds infrequent behaviour filters to all steps of IM [5]. If the model obtained with IM is not precise enough at evaluating the quality criteria (overfitting, underfitting, etc.), the miner is applied again using the same sub-log selecting the IMi. This process is iterated until precise models for all the student profiles are discovered. Finally, the models can be compared through visual inspection. The described process is structured as follows.

1. Input: Sub-log for a student profile
2. Mine using IM ⟶ Discovered model
3. if Discovered model is [Imprecise]
4. Mine using IMi ⟶ Discovered model
5. Go to step 3
6. else //*Discovered model is* [*Precise*]
7. if [Pending student profiles]
8. Go to step 1
9. else //*Not* [*Pending student profiles*]
10. Compare models

E. Conformance Checking

Finally, we aim to support the assessment process according to the performance during the game. Previously discovered models for specific student profiles are the input, along with an event log, for the conformance checking techniques. These techniques make it possible to compare the behaviour of a process model and the behaviour recorded in an event log. Replay is the process to quantify this comparison. It simulates the event log cases given a discovered process model, observing each log trace and showing the logic between the activities in the model.

Our method integrates the assessment process conducting a comparison of performance using assessment models or profiles, an approach proposed by Hainey et al. [6]. Using the discovered model concerning the MSS as the assessment model, the comparison is conducted replaying all the traces of the event log to the MSS model. As a result, the alignment and a fitness value to quantify how each trace (student) fits into the model (most successful students) are obtained.

Fitness is the most suitable criterion for conformance checking techniques as it measures how well the model is able to replay most of the traces in a log. Replay techniques can quantify the fitness, finding an alignment of traces in the event log with the control flow of the process models. The nodes of a model can hold one or more tokens and a set of directed arrows that represent the transition between the nodes. Transitions are enabled as soon as all the nodes connected via an incoming arrow contain a token. While replay progresses, the number of tokens are counted [31]:

Let k be the number of different traces from the aggregated log. For each log trace $i(1 \leq i \leq k)$, let n_i be the number of process instances combined in the current trace, m_i the number of missing tokens, r_i the number of remaining tokens, c_i the number of consumed tokens, and p_i the number of tokens produced during the log replay of the current trace. The token-based fitness metric f is defined as follows.

$$f = \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i\, m_i}{\sum_{i=1}^{k} n_i\, c_i}\right) + \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i\, r_i}{\sum_{i=1}^{k} n_i\, p_i}\right)$$

Trace alignment is established according to the relation between "moves" in the log and "moves" in the model. First, "move on log" represents an event occurring in the log that could not be related to an action in the model. Second, a "move on model" represents the cases where an activity is executed in the model but the log does not contain an event to map to that activity. Third, a "synchronous move" points to an event contained in the log corresponding to an activity executed in the model and vice versa.

Alignment measures the fitness of a trace as a value between 0 and 1. The alignment is maximized with the number of synchronous moves and is minimized with the number of moves on log and moves on model. The value 0 means the poorest fitness between the log and the model while the value 1 corresponds to a perfect fitness, meaning that the alignment only contains synchronous moves. The obtained fitness reflects how the log traces fit to the model and vice versa. In our case, it makes it possible to quantify how much each student's behaviour is similar to the behaviour followed by the MSS.

In this study, a ProM implementation of the replay technique based on trace alignments is used to check the conformance of each log with an assessment model [32]. The described process is structured as follows.

1. Input: Event log, Assessment model
2. Replay ⟶ Report with fitness for all students
3. Export report + Event log ⟶ Enriched event log
4. Classify students by fitness (Enriched event log)

V. Case Study

In the previous section we described the method to support the assessment in a learning experience based on serious games using process mining techniques. The next step is to implement our proposal in an actual educational process. For this reason, a case study in a higher education context has been carried out. This case study was conducted using an action research method: "action research takes its cues – its questions, puzzles and problems – from the perceptions of practitioners within particular, local practice contexts. It builds descriptions and theories within the practice context itself, and tests them through intervention experiments, that is, action research through experiments that bear the double burden of testing hypotheses and effecting some (putatively) desirable change in the situation" [33].

Specifically, action research in education aims to improve an aspect of the research focus, providing practitioners with new knowledge and understanding of how to improve educational practices or resolve significant problems in learning contexts [34]. This goal can be accomplished by examining actions carried out against the original hypotheses. The theory must solve a practical problem and generate knowledge within our context, the assessment process. To this end, a method based on process mining techniques was proposed. The conducted case study provided an event log to be used as input to test our method. Then, their different stages were applied following the considered research questions. Finally, an analysis of the results for each stage was performed, presenting all detected findings.

A. Study Setup

The case study was conducted through an experiment in the "Databases" course, compulsory for the students in the Computer Science degree program in the University of Cadiz (Spain) during the second semester of their second year. Our experience is focused on the skills related to the design of conceptual data specification through an E/R diagram analysing textual requirements. Therefore, the students need to apply the specific skill for relational databases analysis and design focusing on the learning outcome related to the E/R diagram modelling: knowledge of how to produce a logical and conceptual design of a database. The skill and learning outcomes are included in the course syllabus aligned with the ACM/IEEE Computing Curricula recommendations [35].

The study was carried out in a compulsory workshop where students had to play, individually, a serious game specifically designed for the experience [36]. In all, 110 students participated in the workshop. The video game was developed using the Unity engine and can be run on multiple platforms (Windows, GNU/Linux and MacOs).

The player is challenged to design an E/R diagram according to the provided textual requirements. The proposed exercise inside the game is based on the practical example included in the appendix of [37], a widely used reference for database concepts. The E/R diagram to be designed includes 6 entities, 19 attributes, 7 relations and 14 text boxes to insert relation cardinalities.

B. Overview of the Serious Game

At the beginning of the game, a unique identifier for the player is requested. This identifier is included in the event log and used to relate the performed actions with the specific student. Once the identifier is inserted, the player will be allowed to navigate between different screens through a game menu. In addition, an option to confirm the E/R diagram and exit the game is included.

The map screen is the default screen presented to the player. In this screen, the player must collect the textual requirements for the proposed exercise by clicking on the buildings in the campus. When the player selects a building, a Non Player Character (a class delegate, a lecturer or a department head) provides new textual requirements. The notepad screen includes all the textual requirements collected by the player. Each requirement is composed of one or two sentences where the most important words to be considered for the design of the E/R diagram are highlighted in a different colour.

Last but not least, the player must design the E/R diagram using the provided tools in the editor screen, split into an inventory bar and a work panel if needed. An example of an E/R diagram design is shown in Fig. 2. First, the player can choose an element from the

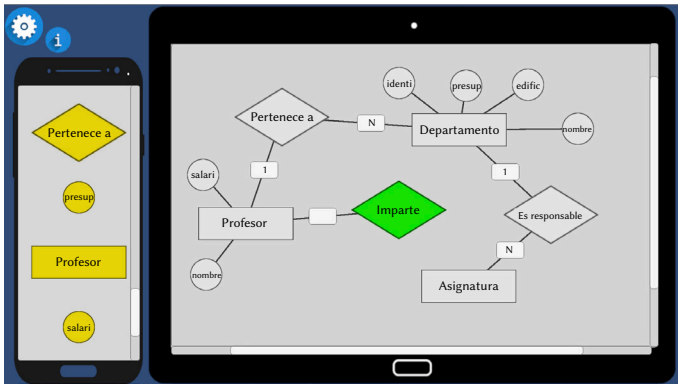

Fig. 2. Example of an E/R diagram being designed in the editor screen (in the Spanish language). The inventory bar is shown on the left and the work zone corresponds to the rest of the screen.

inventory bar and add it to the work panel. Then, the work panel allows the player to organize the elements of the model, relate them, and set their cardinalities (0, 1 or *N*). Finally, the player can easily remove any element of the work panel and add it later as the same or as a different type.

C. Dataset Processing

An event log is created when a player confirms their designed E/R diagram and exits the game. Each event contains: a unique identifier for the player, the performed interaction, a timestamp and a set of additional data according to the type of the interaction. After collecting the resulting event logs for each student, we conducted a processing through the executions of some scripts to join all the event logs into a single dataset. This dataset consisted of 35,931 events. Finally, only those interactions directly related to the design of the E/R diagram were kept, filtering out the other interactions, such as map actions, navigation between screens, etc. An event log with 9,402 events (one per interaction) and 110 traces (one per student) was obtained.

There are 18 types of interactions related to the design of the E/R diagram. These interactions cover the start/end game and the different operations that the player can apply to the different elements of the E/R diagram (entity, attribute and relation): add, delete, link and unlink. In addition, there is an interaction for each value of the cardinality assigned to the union of an entity with a relation (0, 1, *N*) and another to leave it empty.

In-game errors were classified into two categories and seven types, storing the count of each type of error. This classification is based on the usual criteria followed by the course instructor to grade the E/R diagrams designed by the students. Each category represents the level of the error: "major" errors are critical errors and "minor" errors are imprecisions or less important mistakes. On the one hand, 4 types of major errors were considered: fake entity, missing entity, fake relation and missing relation. On the other hand, 3 types of minor errors were considered: fake attribute, missing attribute and fake cardinality. "Fake entities/relations/attributes" refers to E/R diagram elements that should have been designed as another type of element, while "Missing entities/relations/attributes" include those cases where the elements were not designed in the E/R diagram.

In the case of several errors coming from the same origin, only the source error was considered. For instance, if a player missed an entity, only that error was taken into account, ignoring the consequent missed attributes.

The final in-game outcome is automatically calculated in the dataset processing considering the elements (entities, attributes, cardinalities, etc.) of the player's solution that are similar to those of the diagram solution provided in [37]:

Let s be the number of elements of the E/R diagram designed by the player during the game that are similar to those of the E/R diagram solution, and let t be the total number of elements included in the E/R diagram solution. The final in-game outcome is the ratio r defined as follows.

$$r = \frac{s}{t} * 100$$

This outcome does not represent an usual grade, since we aim for a proper interpretation of the textual requirements and a high level of similarity with the solution. For instance, an E/R diagram with only a 50% similarity was not considered as a valid design.

VI. Discussion of Process Mining Techniques for Skill Assessment

The method discussed in the previous sections was implemented using the processed event log described above as the main input object of ProM. We present and discuss the results of implementing our proposal concerning the research questions previously introduced.

A. Research Question 1 – Student Outcomes According to In-Game Errors

In our first stage, we aim to identify the most decisive in-game errors for the outcomes obtained by two specific student profiles, viz., the MSS and the LSS. A classification of the outcomes based on in-game errors was carried out generating decision trees to obtain clusters of students. An initial use case was conducted to filter the log according to the length of the trace and the in-game outcome. The E/R diagram corresponding to the supplied solution has 6 entities, 19 attributes, 7 relations and 14 cardinality inputs: 46 elements in all. In addition, link events are mandatory to properly build the E/R diagram. Therefore, some traces with few events and a poor final outcome regarding quick tests, where the students only explored the game options, were considered as noise in the event log.

An initial analysis of the use case was conducted to identify and filter the cases considered as noise. First, the in-game outcome of the student was selected as the dependent characteristic and the length of the trace as the single independent characteristic. Then, the event selection filter was not used, maintaining all the events of the log. This analysis provided a decision tree with which to classify the in-game outcome for each trace according to the length of the trace.

The decision tree includes nodes labelled with the corresponding independent characteristic, which was "Trace length" for all nodes in this analysis. The nodes are linked with other nodes or with the generated clusters. All links show the corresponding condition for the value of the independent characteristic. The generated clusters are represented as the leaves of the tree, which are named CL_i (with values of i ranging from 1 to *N*). These clusters are generated by the decision tree algorithm.

The clusters are also labelled with the interval of the values included for the dependent characteristic, with the in-game outcome in this and subsequent analyses. In addition, there were some students who were not classified, because their in-game errors and in-game outcomes did not fulfill the requirements of any cluster. Therefore, not all the outcome intervals or values are covered, because the decision tree only displays the students correctly classified in clusters.

Fig. 3 illustrates a section of the obtained decision tree. In this analysis, we focused on Cluster CL1 because it has the least number of events (≤ 78) and also covers the lowest outcomes: [24.05–51.90). This evidences that this cluster includes the traces of students corresponding to limited game experiences. Cluster CL1 contains 12 traces of students, which were considered as noise and removed to

refine the event log. This analysis makes it possible to maintain in the log other students with similar low outcomes but longer traces.

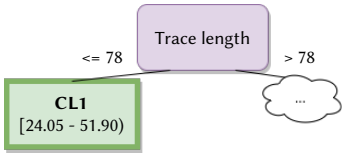

Fig. 3. Decision tree. In-game outcome according to the trace length (CL1).

Once the event log was filtered, how the in-game errors affected the students' outcomes was identified in the next (second) analysis. Again, the in-game outcome of the student was selected as the dependent characteristic. Unlike in the first analysis, several independent characteristics were included, selecting all the possible types of in-game error registered in the event log and listed in the previous section. No event selection filter was used because the log was already filtered according to the results of the first analysis. The obtained decision tree is shown in Fig. 4 and Fig. 5: it has the same structure as that in Fig. 3. In this case, the nodes of the decision tree can have different labels due to the types of error that were selected as the independent characteristics.

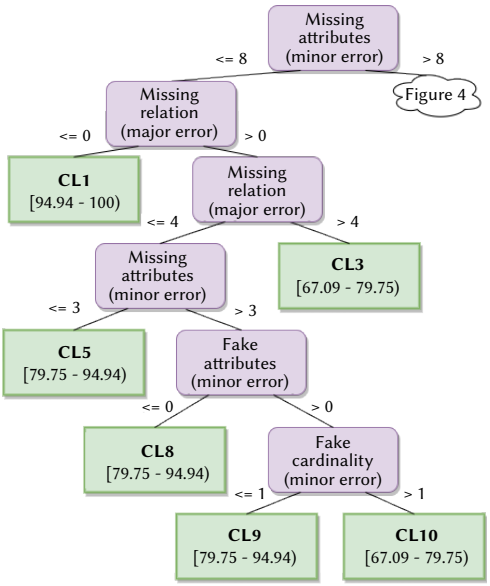

Fig. 4. Decision tree. In-game outcomes based on in-game errors (Left).

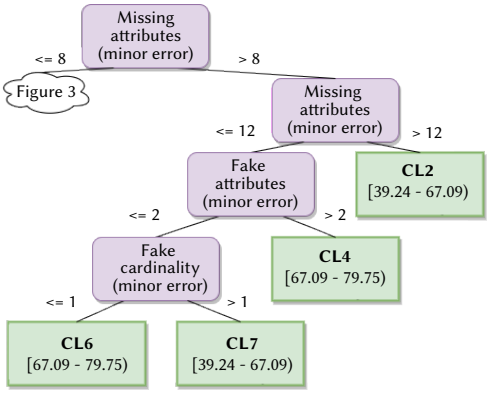

Fig. 5. Decision tree. In-game outcomes based on in-game errors (Right).

Cluster CL1 includes the interval with the highest outcomes [94.04–100), with "missing attribute" and "missing relation" their dependent errors. This cluster includes 15 students who missed less than 9 attributes and included all the relations, achieving a final outcome of at least 94.94 out of 100. In addition, students who achieved an outcome equal to 100 were not classified into this cluster because they did not make any in-game errors.

Then, the three clusters with the next high outcomes (CL5, CL8 and CL9) cover the interval [79.75–94.04) and include 24 students in all. First, Cluster CL5 includes up to four missing relations and three missing attributes at most. Then, Cluster CL8 also includes the cases without fake attributes and increases the number of allowed missing attributes to four. Lastly, Cluster CL9 includes the traces with any number of fake attributes, but at most one fake cardinality.

The next interval, corresponding to results lying within the interval [67.09–79.75), is divided into four clusters (CL3, CL4, CL6 and CL10), which include 20 students in all. These clusters depend on the same type of errors that the previous clusters did, but with higher values. Lastly, the interval with the lowest outcomes is [39.24–67.09), which is covered by two clusters (CL2 and CL7) and includes 24 students in all. Cluster CL2 has cases with more than 12 missing attributes while Cluster CL7 depends on more types of error. In addition, two students achieved a similar outcome but they were not classified into these clusters because they made different in-game errors.

Diverse behaviours could have been performed in the clusters obtained due to their wide intervals of outcomes. Therefore, an iteration of the process to focus on specific profiles and obtain more detailed results was conducted. Based on the previous clusters, two sub-logs were created to analyse the successful (CL1) and the less successful students (CL2 and CL7). First, the sub-log corresponding to the successful students includes those students with a final outcome greater than or equal to 94.94, composed of 17 traces and 1,521 events. Then, the sub-log corresponding to less successful students includes students with outcomes less than or equal to 67.09, composed of 26 traces and 2,261 events. The decision trees obtained by these views are shown in Fig. 6 and Fig. 7 and have the same structure as the previous ones.

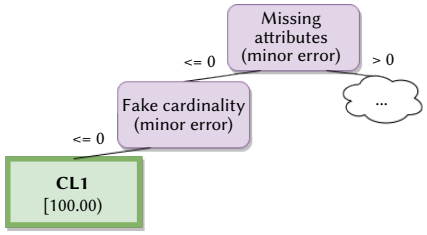

Fig. 6. Decision tree. In-game outcomes for successful students based on in-game errors (CL1).

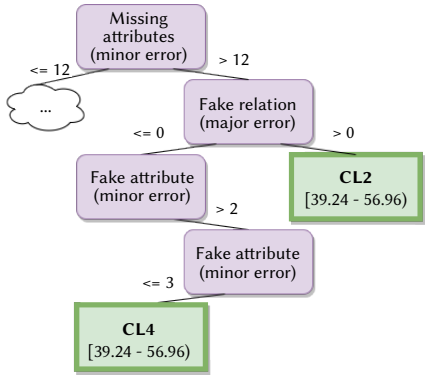

Fig. 7. Decision tree. In-game outcomes for less successful students based on in-game errors (CL2 and CL4).

On the one hand, Fig. 6 shows that Cluster CL1 is labelled with [100.00] as it only includes a single value instead of an interval. In this case, this cluster contains the two students who achieved the maximum

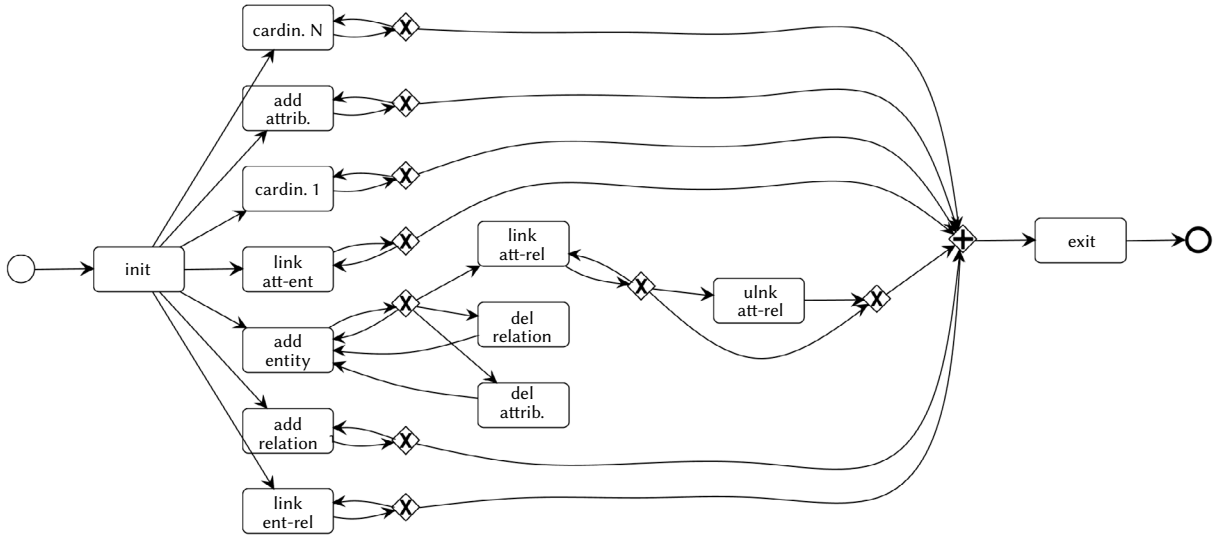

Fig. 8. Most successful students (MSS) including all paths.

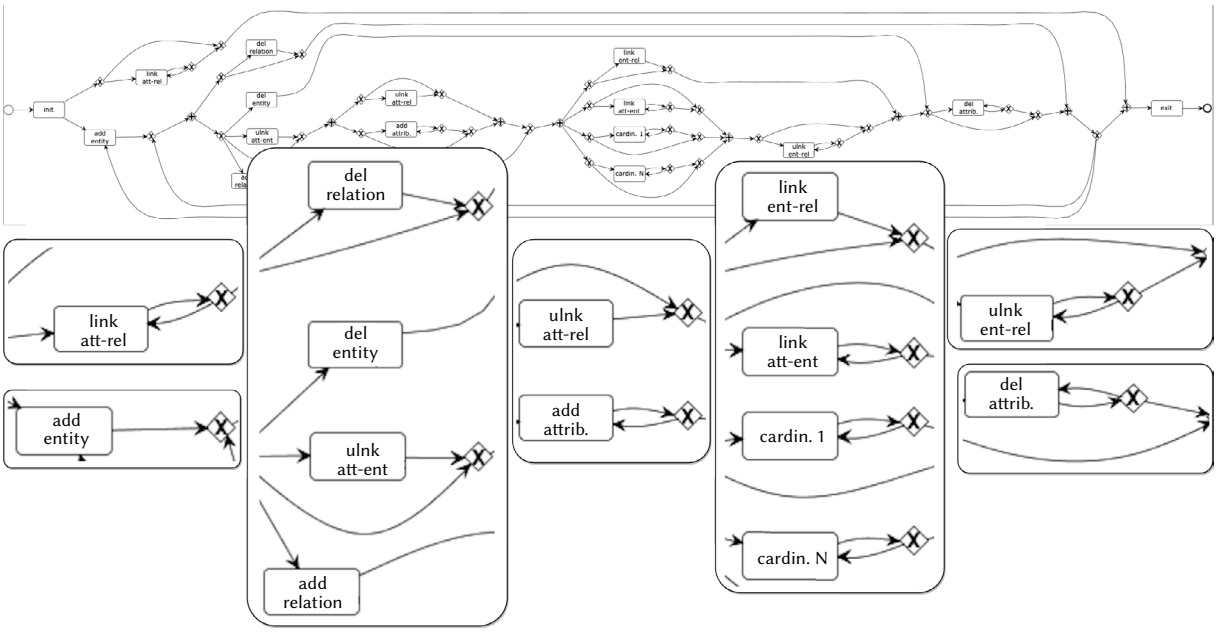

Fig. 9. Least successful students (LSS) including all paths.

outcome in the game. The only errors related to this outcome are missing attributes and fake set cardinalities. Therefore, only those students who properly set the attributes and the cardinalities achieved the maximum outcome. These two students are considered as the MSS. On the other hand, Fig. 7 shows two clusters (CL2 and CL4), which cover the interval [39.24–56.96). In these cases, more than 12 attributes were wrongly set. In addition, these students made fake relations or included up to three fake attributes. The log also includes another student with a similar outcome (51.90) but not classified in the cluster due to different in-game errors. All these students (8) were considered as the LSS.

Based on their in-game errors, the conducted process showed the justification for why a student was included in the respective cluster. According to the results of the previous decision trees, missing attributes was the most decisive type of error to achieve successful outcomes because it is in the top of the three trees. The fake relation type of error also discriminates between the successful and less successful students. Lastly, fake cardinalities and fake attributes were also critical to discriminating between the MSS and the LSS, respectively. These types of error were considered as the most decisive in-game errors for the outcomes achieved by the MSS and the LSS, answering RQ1.

B. Research Question 2 – Behaviour Analysis Between Student Profiles

After identifying the most decisive in-game errors for the MSS and LSS, we aim to apply model discovery techniques to detect similarities and differences in the performance of these student profiles. According to previous results, the corresponding sub-logs from the clusters including MSS and LSS are obtained: [100] and [39.24–56.96), respectively. First, the MSS sub-log includes two traces and 171 events, corresponding to those students who achieved the maximum outcome in the game (100). Second, the LSS sub-log includes 8 traces and 691 events, representing the lowest results after excluding traces considered as noise due to a scarce trace length.

Firstly, the event sub-logs were loaded in ProM to carry out the model discovery. In Business Process Model Notation (BPMN), empty circles represent the starting and the ending states. Then, all the nodes are

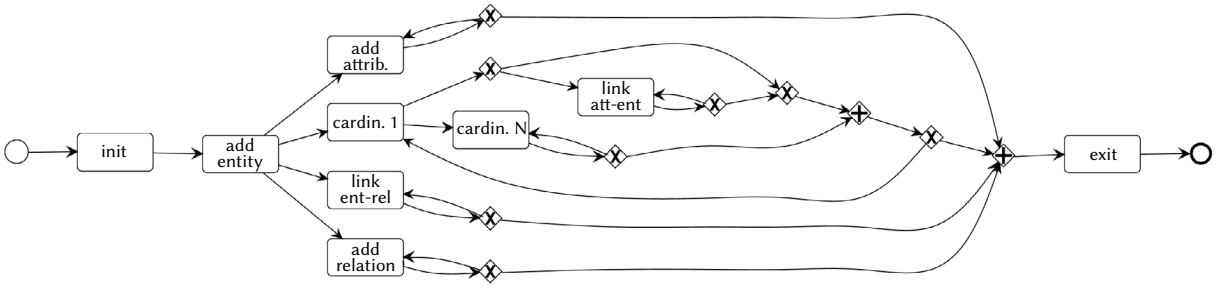

Fig. 10. Most successful students (MSS) filtering out less frequent paths

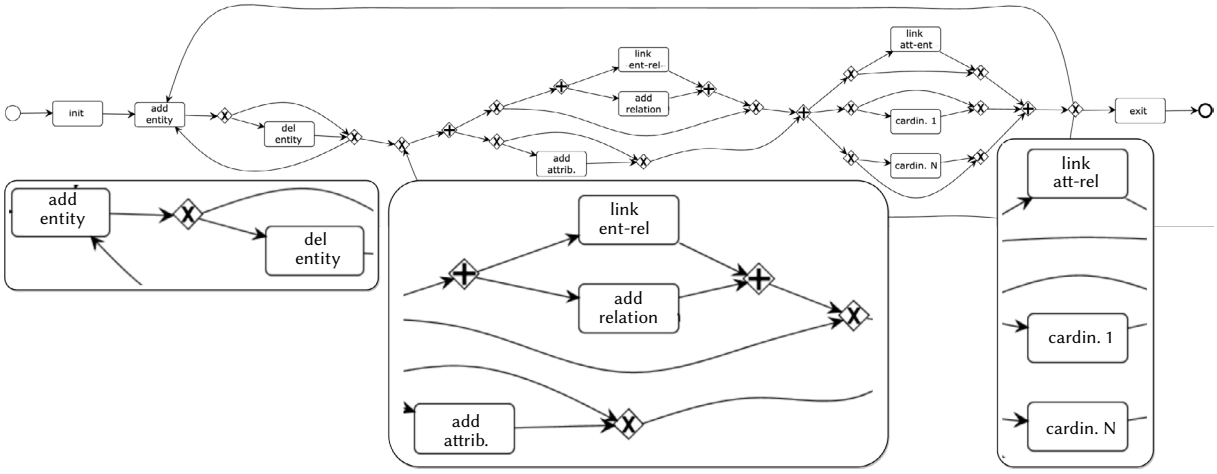

Fig. 11. Least successful students (LSS) filtering out less frequent paths.

labelled with an event action, which in this context corresponds to one type of player interaction. These nodes are linked by directed arrows to represent the sequentiality. In addition, two types of diamonds are present to control the flow. Diamonds with a "+" represents a join for input paths and a parallel split of output paths to follow all the subsequent gateways. Diamonds with a "X" corresponds to decision points where the path bifurcates in case of having multiple outputs. The names of the events have been shortened to improve the visualization.

In the first iteration, models for the MSS and the LSS profiles were generated by using IM, as shown in Fig. 8 and Fig. 9, respectively. The wide variety of sequences are reflected in the model for MSS (Fig. 8) through numerous loops, enabling practically arbitrary executions of activities. In addition, all the types of event included in the sub-log (13), even the least frequent ones, are present in the model. The visual inspection was supported by checking the events included in the sub-log, which confirmed that any traces started with interactions such as link-ent, link-rel, cardin. 1 or cardin. *N*; but the model made this behaviour possible. Therefore, the resulting model is too imprecise, being underfit and enabling too many behaviours, thus it does not provide a good reference for the students' behaviour. Similar problems are evidenced in the model for LSS (Fig. 9). Although this model does not present as many arbitrary loops as the model for MSS, it presents a too complex structure and too many paths to skip the majority of activities of the process. Finally, it also presents all the types of events included in the sub-log (16), which can result in the inclusion of infrequent event activities.

Considering these issues, IMi was implemented in the next iterations for the MSS and the LSS profiles with different incremental values for the noise threshold. The use of IMi makes it possible to select a noise threshold from 0.00 to 1.00, where setting 0.00 guarantees a perfect log fitness.

Searching for a balance between precision and generalization ability, less frequent paths were filtered by setting a 20% noise threshold. The resulting models for MSS and LSS are shown in Fig. 10 and Fig. 11, respectively. Compared with the previous ones, both models are more suitable. These models are more precise and do not include the less frequent types of events: the MSS model includes 9 instead of 13 and the LSS model includes 10 instead of 16. Therefore, infrequent paths were also filtered out. In addition, the models have a simpler structure, easier to understand and more significant.

The discovered models for the MSS and the LSS profiles still have several loops even though the infrequent paths were filtered out. In essence, the design of an E/R diagram is carried out through multiple loops for the different types of actions, such as adding elements and creating links. Therefore, we assume that the occurrence of these loops is a consequence of the iterative nature of the analysed process.

Significant differences in the structure between the MSS and LSS models can be seen (Fig. 10 and Fig. 11, respectively). One the one hand, the MSS model includes the add entity interaction and several small loops for the rest of the interactions, so many paths are possible. However, the setting of the cardinalities and linking attributes are in the same path. This sequentiality could suggest that these types of interactions were usually done in sequence. This is a recommended pattern in the design of E/R diagrams, as both aspects are deeply interrelated. On the other hand, the LSS model has two big loops and a more linear structure, with three subprocesses. First, there are the add and delete entity interactions. This similarity with the MSS model makes sense due to fact that the entities can be considered as the starting point of the design of an E/R diagram. Second, attributes and relations are added. In addition, this subprocess includes the link between entities and relations. Third, cardinalities are set and attributes with entities are linked. This behaviour is similar to that performed by the MSS because these interactions were grouped as well. In general, this structure suggests a common heterogeneity in the behaviour performed by the LSS but with differences because of the multiple bifurcations the model presents.

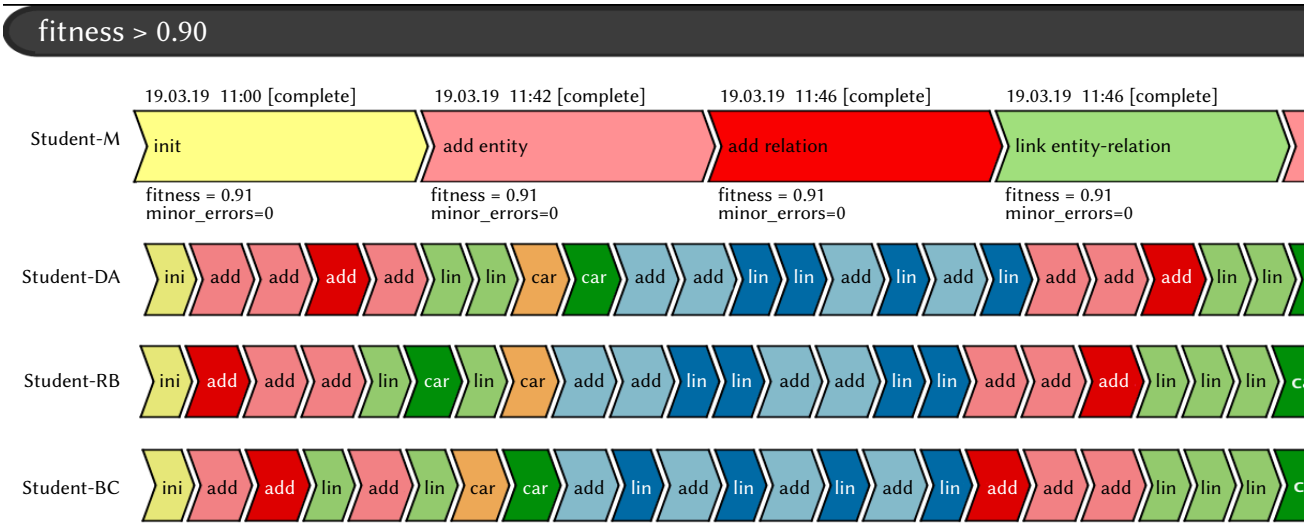

Fig. 12. Traces of students who obtained a fitness > 0:90.

The MSS model only includes interactions that add value to the design of the E/R: there are no removing interactions such as deleting elements or unlinks. Although its corresponding general model (Fig. 8) included some of these types of interactions, they were filtered out once the infrequent paths were removed. However, the model for LSS (Fig. 11) still includes one interaction of this type after filtering out the infrequent paths: "delete entity". This behaviour evidenced an indecision at the beginning of the process. The creation of entities is the most basic step in the design of an E/R diagram, so the performance also evidences some limitations to the use of design skills. In addition, the MSS model shows the students added all the entities at the beginning while the LSS model includes loops to return to this point and open the path to new deletions of entities. This difference in the behaviour could be evidence that the indecision continued in later stages of the design process.

The lack of other removal interactions, such as deletes or unlinks, in the LSS model could suggest that the hesitations were only focused on interpreting the requirements to detect what elements should be designed as entities. This is supported by checking the event log, observing that less than 5% of the events performed by the LSS were removal interactions other than delete entities. However, as discussed in Section V.B, the serious game only considered the source error when several errors come from the same origin. Therefore, the LSS could have made consequent errors and had hesitations interpreting the whole of the requirements.

Except for "delete entity", the models for MSS and LSS include the same types of interaction. Although their general models (Fig. 8 and Fig. 9) present more variations between them, the majority of these variations were removed from the models when filtering out infrequent paths. This similarity can be evidence that the LSS usually avoided infrequent or not required interactions, such as "link attribute-relation" and "cardinality 0".

According to the discovered models, the results show that there are some similarities and differences in the performance between the MSS and the LSS. All these similarities and differences answer aÿrmatively the RQ2, in that both profiles follow similar behaviours but with differences in some key aspects.

C. Research Question 3 – Student Classification According to Performance

The aim of this work is to support student assessment through a behaviour comparison using an assessment model or profile. This comparison was conducted applying a conformance checking to replay the traces of all students to the MSS model discovered in the previous subsection. This process provided the fitness for each trace according to the MSS model, quantifying the similarity between each student and the MSS behaviour. In this process, the MSS model previously discovered after applying IMi to ignore infrequent events and paths was used.

First, the whole event log and the MSS model were loaded in ProM to be used as input in a plugin called "Replay a Log on Petri Net for Conformance Analysis", an implementation that can use several algorithms to replay traces over a model. This plugin was selected because it yields detailed results, providing specific fitness values for each trace and making it possible to export them in an external file. In this study, we used the Dijkstra and ILP algorithms, which provided similar results. This plugin was used to replay the log and calculate the fitness values according to the alignment between the log and the model. As we discussed during the presentation of the conformance checking stage of our method in Section IV.E, this process provides a conformance checking report that can be exported and processed to extract the fitness values. Then, the original event log was enriched by including the corresponding fitness in each student's trace. Students with similar in-game outcomes could have carried out different behaviours, so the fitness could be different as well. Finally, the enriched event log was loaded in ProM to be explored and classify the traces.

Aiming to compare the in-game outcomes and behaviours (fitness), the average value for the fitness of those who share the same in-game outcome was calculated. Finally, the in-game outcome and fitness were associated, employing the Pearson correlation coeÿcient, obtaining 0.75. We can aÿrm that the students' behaviour and their achieved in-game outcomes have a reliable correlation in the context of this case study.

ProM enables filtering traces according to the fields included in the event log. Fig. 12 shows all the students' traces with a fitness higher than 0.90. The different traces are shown in independent lines and include the sequence of events. The traces also contain the unique identificator for each student, anonymized as "Student-" plus one or more capital letters. This information was provided to the students who fulfilled the filter requirements. Each trace is composed of small boxes that correspond to the events of the corresponding student. The events are labelled with the shortened name and are displayed using different colours for each type of event. For instance, "init" is shortened to "ini" and displayed in yellow.

The conformance checking stage of our method provides an interactive chart to explore the traces and events. In Fig. 12, we selected the first student (Student-M) to expand their events and show the specific dates and times of each event. A sample of the included fields is displayed under the trace but all the fields with their types and values can be explored after the traces. For instance, "Student-M" is part of the MSS students because that student achieved an in-game outcome of 100, that is, no errors were made.

This process can be iterated using different fitness values and obtaining the corresponding students. The traces of students with a fitness between 0.90 and 0.80 were obtained. This provided 82 traces, meaning a high ratio of the traces included in the whole event log (74.54%). This evidences a high similarity in the behaviour of the majority of the students. To obtain more detailed results, a shorter range for the fitness value was used for the filter (0.90 and 0.89). This filter provided 10 traces. After exploring these data, we found that 8 of these students achieved in-game outcomes higher than 89.00, showing a positive correlation with their fitness. In addition, the remaining two students (Student-FD and Student-QC) achieved unusual in-game outcomes (51.90 and 67.09, respectively). Although those two students achieved lower results, they showed a similar behaviour to the others with high in-game outcomes. Therefore, unlike assessment methods based only on the obtained grade, this assessment method considered behavioural evidence to detect how the students applied their skills.

Beyond the Pearson correlation coeÿcient previously calculated (0.75), the course's supervisor provided positive feedback about being supported by the proposed assessment method. This provides objective evidence about the behaviour followed by the students during the experience. The obtained fitness value allowed the supervisor to detect students who, despite carrying out proper behaviour during the process, did not have this reflected in the grade obtained. Multiple factors could be involved in this result, i.e. good skills about E/R designing but a lack of requirements analysis. Therefore, the supervisor used this assessment method to optimize the revision process, revising only specific cases where discrepancies between grades and behaviour were found.

In accordance with the conducted conformance checking, the replay report provided a fitness value for each student trace. This fitness quantifies the similarity between one student's behaviour and the assessment model. It was incorporated in the event log as an additional field, so all the students could be classified according to their performance during the game. Therefore, the employed filters provided a scalable support for the students' assessment through a behaviour comparison using assessment models, thus answering RQ3.

D. Threats to Validity

After using the proposed method in a case study, it is essential to identify potential threats to its validity that may occur in the development of the study. This subsection of the paper discusses possible construct threats to the validity of the experiment and our proposal to measure/mitigate them.

Some of the configurations used during the process could be coupled to the analysed dataset, such as the selected noise threshold to filter out infrequent paths. This issue was considered as a challenge to internal validity. All the analysed data to replicate the experiment and additional figures to detail the results are available in an open multimedia repository [36]. This information ensures the reliability of this study and our results.

The limitations of assessment processes discussed in the paper were detected in studies where the assessment of the learning process was focused on the acquisition of skills. As we have not reviewed any learning processes with assessment focused on getting knowledge, we can not evaluate if our method can be adapted or not for those learning processes. This limitation was considered as a challenge to the external validity. A literature review, additional modeling and empirical research are necessary to confirm if learning processes with assessment focused on getting knowledge present the same limitations.

Our proposal has two requirements. First, the student has to resolve the game by applying one or more specific skills and, second, the game has to provide logs with relevant information for the assessment. The adaptation of the method in other skill assessment problem is dependent on how well the game helps/requests the development of the skill from the students, and the quantity and quality of the information stored in the logs. Once a serious game provides logs with relevant information to be considered in the skill assessment, the proposed method allows to easily analyze in detail thousands of events produced during the game experience. These requirements were considered as a challenge to the external validity.

Although the proposed method is generic and can be used in diverse contexts, the results of this paper are limited to the scope of the conducted case study. All the configurations employed during the process depend on the skill to be assessed and the information stored in the dataset. For instance, if we want to assess a time-focused skill in resolving a task assignment, we could obtain the models considering the time invested in the game experience instead of the in-game errors. Additionally, the dataset could be filtered according to a minimum in-game outcome to only consider those students who successfully passed the assignment.

In order to generalise the findings to other processes, a replication of the experiment in other learning experiences is needed.

Regarding the use of our method in other database courses, the assessed skill in this paper is included in the course syllabus aligned with the ACM/IEEE Computing Curricula recommendations [35]. It is a widely used reference for higher education computer science programs, so the proposed method should adjust as well to a database course other than the one taken as source for this particular study. Since the presented case study was conducted in a single database course, the limitations previously discussed should be considered as well.

The assessment was supported in an experience based on a serious games, through process mining techniques. The learning experience aims to assess a skill for the analysis and design of relational databases, focussing on the design of E/R diagrams. However, this kind of process has an intrinsically iterative essence because the same events are performed multiple times during the game. This feature resulted in process models with multiple loops and bifurcations even after filtering out infrequent paths. Addressing this iterative essence, applying process mining techniques, was a challenge to the construction validity.

VII. Conclusions

As learning processes are focused on the acquisition of skills, students must be assessed according to their level of proficiency in these skills. In this paper, we aim at supporting a solution for skill assessment through behaviour comparison, using assessment models. In order to validate the proposal, a case study was conducted in a course on databases, part of a Computer Science degree program. More than 100 students had to apply database analysis skills by designing an E/R diagram during a serious game. In all, the interactions of the students provided 35,931 events, which were processed and refined to 9,402 events. The main research question is whether process mining techniques can support scalable assessment in a learning experience based on serious games. In order to answer this question, it was divided into three research questions.

In the first question (RQ1), we aimed at identifying the most decisive in-game errors for specific student profiles: the most successfull students (MSS) and the least successfull students (LSS). A clustering of dynamic behaviour through decision trees was employed to generate clusters and classify the students. How in-game errors affect the students' outcomes was identified in an analysis of a use case. First, having missing attributes was the type of error that was most decisive in failing to achieve successful outcomes, because it is in the top of the majority of the trees obtained. Then, another type of error was also critical for differentiating between successful and less successful students (fake relations). Additional errors differentiated the MSS from the rest of the successful students (fake cardinalities) and differentiated the LSS from the other students with poor results (fake attributes).

In the second question (RQ2), we aimed at detecting similarities and differences between the performances of specific student profiles. Based on previous analysis, models for the MSS and the LSS were obtained applying model discovery techniques through inductive mining. The results showed some similarities and differences in the performances of both student profiles. First, we detected similarities in the type of interactions, iterative performance, starting point for the E/R design, and subprocess for specific interactions (cardinalities and link attribute-entity). Second, the models had differences in their main structure, different types of loops, differences in the occurrence of bifurcations and of interactions to delete entities in the LSS model.

In the third question (RQ3), we aimed at applying process mining techniques to assess the students according to their performance during the game. We proposed integrating the assessment process based on the comparison of the performance using assessment models or profiles. Therefore, the previously discovered model for the MSS was used as the assessment model to replay the whole event log for a conformance checking. Conformance checking supported the inferences from a visual inspection of the process models. The replay provided a fitness value for each student trace, measuring the similarity of the behaviour of each student to the MSS. The obtained fitness and the in-game outcomes presented a reliable and positive correlation in this context. Finally, the fitness was used to enrich the event log and classify students according to their performance.

The proposed method in this paper has several similarities with other automated assessment methods reviewed in this paper. Although there is no formal relation between indicators and evidence, a visual inspection of the discovered models was made that focused on detecting patterns and evidence of the applied skills during the game. As a stealth assessment method, we also collected data during the game to gather evidence through a quiet process, removing test anxiety as much as possible. More specifically, in this case study, the design of an E/R was an iterative process that can produce a large quantity of interactions for which a manual analysis is not feasible. Process mining techniques proved to be a suitable solution to handle them.

Unlike other data-centred automated assessment methods, process mining is a mixed approach, lying between data-centred and process-centric. The proposed method can model students' behaviour and discover knowledge from event data. First, this mixed approach makes it possible to use event data for detecting the most decisive in-game errors and how they determine the students' outcomes. Second, the process-centred techniques are used to model students' behaviour and compare their performance.

In addition, other automated assessment methods are usually focused on a formative aim, specially in intelligent tutoring systems. We aimed at a summative assessment purpose, but focusing on the behaviour performed during the complete game experience instead of only the final results. This additional feedback allowed the supervisor to detect students who, despite carrying out a proper behaviour during the process, did not have this reflected in the grade obtained.

The scope of the validation for the proposed method is limited to a single skill of "analysis and design of relational databases". However, the method provides techniques to assess other skills because it makes it possible to use several assessment models or profiles, i.e. students with the highest outcomes in specific features. Therefore, independent assessment models can be defined as long as the skills need to be assessed independently.

Although we used an assessment model discovered from the analysed data set (i.e. students with the highest in-game outcomes), this model could have been previously defined by an expert in the area of application (i.e. databases), for instance, by using a model that represents the expert's behaviour, or by using external tools.

The fitness values obtained in the replay represent the indicators to be considered to assess the behaviour of students during the game. Fitness measures the gap between the behaviour represented by the assessment model and the behaviour of each student. An implication to take into account so as to put the method into practice is that fitness values have to be exported from the conformance checking report. Using the fitness, filters were applied to detect the level of similarity that students achieved with the "expert" behaviour. This fitness value could also be used as an input in additional methods.

This paper provides a methodological contribution to the use of process mining techniques to support skill assessment in serious games. Other skill assessments based on process mining techniques for learning experiences based on serious games have not been identified in the literature. In addition, the serious game used in the case study is a software contribution specifically developed and aligned with the assessed skill: the design of a conceptual data specification through an E/R diagram analysing textual requirements. Other serious games with the same purpose have not been identified in the literature.

As the main conclusion, process mining techniques can support a scalable assessment method in learning experiences based on serious games. Applying the process mining model discovery was suitable for analysing the behaviour of students in sequential E/R diagram modelling processes. The tools used provided an automated support to assessing the developed skills during the gameplay. Therefore, we consider the previous evidence as positive to answer the research question of this study.

Regarding future work, we have in view extending the developed serious game with new features and improvements to enable the assessment of other skills from the "Databases" course, beyond the design of a conceptual data specification through an E/R diagram analysing textual requirements. Another line of research would involve applying the method to different types of learning experiences to study learning processes in different contexts. Finally, an additional line of research would be focused on the teaching-learning process, comparing the results obtained by the students with the teacher's assessment.

Acknowledgments

This paper is part of the R&D project CRÊPES (ref. PID2020-115844RB-I00), funded by MCIN / AEI doi:10.13039/501100011033/. We thank Gregorio Rodríguez Gómez for his advice on assessment processes background.

References

[1] D. Djaouti, J. Alvarez, J.-P. Jessel, O. Rampnoux, "Origins of Serious Games," in *Serious Games and Edutainment Applications*, M. Ma, A. Oikonomou, L. C. Jain Eds., London: Springer London, 2011, pp. 25–43, doi: 10.1007/978-1-4471-2161-9_3.

[2] T. M. Connolly, E. A. Boyle, E. Macarthur, T. Hainey, J. M. Boyle, "A systematic literature review of empirical evidence on computer games

and serious games," *Computers & Education*, vol. 59, pp. 661–686, 2012, doi: 10.1016/j.compedu.2012.03.004.

[3] J. A. Caballero-Hernández, M. Palomo-Duarte, J. M. Dodero, "Skill assessment in learning experiences based on serious games: A Systematic Mapping Study," *Computers and Education*, vol. 113, pp. 42–60, oct 2017, doi: 10.1016/j.compedu.2017.05.008.

[4] G. Siemens, S. Dawson, G. Lynch, "Improving the Quality and Productivity of the Higher Education Sector Policy and Strategy for Systems-Level Deployment of Learning Analytics," Society for Learning Analytics Research, 2013.

[5] W. van der Aalst, A. Adriansyah, A. K. A. De Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. Van Den Brand, R. Brandtjen, J. Buijs, A. Burattin, J. Carmona, M. Castellanos, J. Claes, J. Cook, N. Costantini, F. Curbera, E. Damiani, M. De Leoni, ... M. Wynn, "Process mining manifesto," in *Lecture Notes in Business Information Processing*, vol. 99 LNBIP, 2012, pp. 169–194, Springer Verlag.

[6] T. Hainey, T. M. Connolly, Y. Chaudy, E. Boyle, R. Beeby, M. Soflano, "Assessment integration in serious games," in *Psychology, Pedagogy, and Assessment in Serious Games*, IGI Global, nov 2013, pp. 317–341, doi: 10.4018/978-1- 4666-4773-2.ch015.

[7] M. Allen, *Assessing academic programs in higher education*. Bolton, MA: Anker, 2004.

[8] V. J. Shute, "Stealth Assessment in Computer-Based Games To Support Learning," in *Computer Games and Instruction*, S. Tobias, J. D. Fletcher Eds., Cambridge: MIT Press, 2011, ch. 20, pp. 503–524.

[9] R. J. Mislevy, G. D. Haertel, "Implications of evidence- centered design for educational testing," *Educational Measurement: Issues and Practice*, vol. 25, no. 4, pp. 6–20, 2006, doi: https://doi.org/10.1111/j.1745-3992.2006.00075.x.

[10] R. J. Mislevy, R. G. Almond, J. F. Lukas, "A brief introduction to evidence-centered design," *ETS Research Report Series*, vol. 2003, no. 1, pp. i–29, 2003, doi: 10.1002/j.2333-8504.2003.tb01908.x.

[11] V. J. Shute, M. Ventura, M. Bauer, D. Zapata-Rivera, "Melding the power of serious games and embedded assessment to monitor and foster learning," in *Serious games: Mechanisms and effects*, U. Ritterfeld, M. J. Cody, P. Vorderer Eds., Routledge, 2009, pp. 295–321.

[12] V. J. Shute, Y. J. Kim, "Formative and Stealth Assessment," in *Handbook of Research on Educational Communications and Technology*, J. M. Spector, M. D. Merrill, J. Elen, M. J. Bishop Eds., New York, NY: Springer New York, 2014, pp. 311–321, doi: 10.1007/978-1-4614-3185-5_25.

[13] A. Mitrovic, M. Mayo, P. Suraweera, B. Martin, "Constraint-Based Tutors: A Success Story," in *Engineering of Intelligent Systems*, Berlin, Heidelberg, 2001, pp. 931–940, Springer Berlin Heidelberg.

[14] P. Suraweera, A. Mitrovic, "An Intelligent Tutoring System for Entity Relationship Modelling," *International Journal of Artificial Intelligence in Education*, vol. 14, pp. 375–417, 2004.

[15] L. Zhuhadar, S. Marklin, E. Thrasher, M. D. Lytras, "Is there a gender difference in interacting with intelligent tutoring system? Can Bayesian Knowledge Tracing and Learning Curve Analysis Models answer this question?," *Computers in Human Behavior*, vol. 61, pp. 198–204, 2016, doi: https://doi.org/10.1016/j.chb.2016.02.073.

[16] M. V. Yudelson, K. R. Koedinger, G. J. Gordon, "Individualized Bayesian Knowledge Tracing Models," in *Artificial Intelligence in Education*, Berlin, Heidelberg, 2013, pp. 171–180, Springer Berlin Heidelberg.

[17] W. M. P. van der Aalst, *Process Mining Data Science in Action*. Berlin Heidelberg: Springer, 2nd ed. ed., 2016.

[18] K. Engelmann, M. Bannert, "Analyzing temporal data for understanding the learning process induced by metacognitive prompts," *Learning and Instruction*, p. 101205, 2019, doi: https://doi.org/10.1016/j.learninstruc.2019.05.002.

[19] H. A. V. D. Berg, "Occam's razor: From ockham's via moderna to modern data science," *Science Progress*, vol. 101, no. 3, pp. 261–272, 2018, doi: 10.3184/003685018X15295002645082.

[20] M. de Leoni, W. M. van der Aalst, M. Dees, "A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs," *Information Systems*, vol. 56, no. July, pp. 235–257, 2016, doi: 10.1016/j.is.2015.07.003.

[21] A. Bogarín, R. Cerezo, C. Romero, "A survey on educational process mining," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, p. e1230, jan 2018, doi: 10.1002/widm.1230.

[22] A. Bolt, M. de Leoni, W. M. P. van der Aalst, P. Gorissen, "Exploiting Process Cubes, Analytic Workflows and Process Mining for Business Process Reporting: A Case Study in Education," in *International Symposium on Data-driven Process Discovery and Analysis (SIMPDA)*, Vienna, Austria, 2015, pp. 33–47.

[23] J. A. Caballero-Hernández, A. Balderas, M. Palomo- Duarte, P. Delatorre, A. J. Reinoso, J. M. Dodero, "Teamwork assessment in collaborative projects through process mining techniques," *International journal of engineering education*, vol. 36, no. 1, pp. 470– 482, 2020.

[24] P. Reimann, "Time is precious: Variable- and event- centred approaches to process analysis in CSCL research," *International Journal of Computer-Supported Collaborative Learning*, vol. 4, no. 3, pp. 239–257, 2009, doi: 10.1007/s11412-009-9070-z.

[25] M. Bannert, P. Reimann, C. Sonnenberg, "Process mining techniques for analysing patterns and strategies in students' self-regulated learning," *Metacognition and Learning*, vol. 9, no. 2, pp. 161–185, 2014, doi: 10.1007/s11409-013-9107-6.

[26] P. Reimann, L. Markauskaite, M. Bannert, "e-Research and learning theory: What do sequence and process mining methods contribute?," *British Journal of Educational Technology*, vol. 45, no. 3, pp. 528–540, 2014, doi: 10.1111/bjet.12146.

[27] H. M. W. Verbeek, J. C. A. M. Buijs, B. F. Van Dongen, W. M. van der Aalst, "ProM: The Process Mining Toolkit," in *International Conference on Business Process Management Demonstration Track*, Hoboken, New Jersey, 2010, pp. 34–39.

[28] V. U. Kumar, A. Krishna, P. Neelakanteswara, C. Z. Basha, "Advanced prediction of performance of a student in an university using machine learning techniques," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, 2020, pp. 121–126.

[29] S. A. Alasadi, W. S. Bhaya, "Review of data preprocessing techniques in data mining," *Journal of Engineering and Applied Sciences*, vol. 12, no. 16, pp. 4102–4107, 2017, doi: 10.3923/jeasci.2017.4102.4107.

[30] S. J. J. Leemans, D. Fahland, W. M. P. van der Aalst, "Discovering Block-Structured Process Models from Event Logs Containing Infrequent Behaviour," in *Business Process Management Workshops. BPM 2013. Lecture Notes in Business Information Processing, vol 171.*, N. Lohmann, M. Song, P. Wohed Eds., Springer, Cham, 2014, pp. 66–78, doi: 10.1007/978-3-319-06257-0_6.

[31] A. Rozinat, W. M. P. van der Aalst, "Conformance checking of processes based on monitoring real behavior," *Information Systems*, vol. 33, no. 1, pp. 64–95, 2008, doi: https://doi.org/10.1016/j.is.2007.07.001.

[32] W. van der Aalst, A. Adriansyah, B. van Dongen, "Replaying history on process models for conformance checking and performance analysis," *WIREs Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 182– 192, 2012, doi: 10.1002/widm.1045.

[33] C. Argyris, D. A. Schön, "Participatory action research and action science compared: A commentary," *American Behavioral Scientist*, vol. 32, no. 5, pp. 612–623, 1989.

[34] G. E. Mills, *Action research: A guide for the teacher researcher*. Boston: Pearson, 4th ed., 2011.

[35] ACM, IEEE, "Computer Engineering Curricula 2016," ACM, IEEE, 2016.

[36] J. A. Caballero-Hernández, "Supporting skill assessment in learning experiences based on serious games through process mining techniques," 2020. doi: 10.6084/m9.figshare.c.4916412.

[37] A. Silberschatz, H. F. Korth, S. Sudarshan, *Database system concepts*. New York: McGraw-Hill, 6th ed. ed., 2011.

Juan Antonio Caballero-Hernández

Juan Antonio 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. Beyond the academic environment, he has worked in different positions in IT, such as web development and managing teams.

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, where he works as an Associate Professor. He is the author of more than 20 papers published in indexed journals and more than 30 contributions to international academic conferences about learning technologies, serious games and the collaborative Web.

Juan Manuel Dodero

Juan Manuel Dodero is Full Professor of Computer Science in the University of Cadiz, Spain. He has a Computer Science degree from the Polytechnic University of Madrid and a PhD degree from the Carlos III University of Madrid. His main research interests include Web science and engineering and technology-enhanced learning, fields in which he has co-authored numerous research papers in international journals and conferences.

Dragan Gašević

Dragan Gašević is Distinguished Professor of Learning Analytics in the Faculty of Information Technology and Director of the Centre for Learning Analytics at Monash University. He served as the president (2015–2017) of the Society for Learning Analytics Research (SoLAR). A computer scientist by training and skills, Dragan considers himself a learning analyst who develops computational methods that can shape next-generation learning technologies and advance our understanding of self-regulated and collaborative learning. Dragan is a (co-) author of numerous research papers and books and a frequent keynote speaker.