2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC) | 978-1-6654-8810-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/COMPSAC54236.2022.00027

SQL Murder Mystery: a serious game to learn querying databases

Lorenzo Canale, Laura Farinetti
Politecnico di Torino
Torino, Italy
(lorenzo.canale, laura.farinetti)@polito.it

***Abstract*—What is serious? What is funny? Who is a player? Who is a student? But, most importantly, who is the murderer? Serious games are gaining an ever increasing interest in education and training, and recent studies have used board games as inspirational. This study introduces SQL Murder Mystery, a serious game inspired by the popular board game Cluedo. This game has been designed to assess students' SQL skills and has been tested in a university database management system course during a lab session in which students played in teams. Query logs were examined to explore the behavioural patterns of the teams, by distinguishing different categories of queries: *exploratory*, *focused*, *review* queries, and to relate behaviour with specific SQL learning goals. The analysis revealed that success in the game and fulfilment of SQL learning goals are correlated. In addition, the game helped the instructors to identify the major knowledge gaps of the students, to allow on-time recovery.**

***Index Terms*—Edutainment, Serious games, Playful learning, Learning analytics, Educational data mining**

I. Introduction

A *serious game* or *applied game* is a game designed for a primary purpose other than pure entertainment [1]. In the educational field, "serious games are games designed in which education (in its various forms) is the primary goal, rather than entertainment." [2]. [3] define six key concepts proper to educational serious game: scenario, enthusiasm, interactivity, user's traces, scoring and learning.

The use of games in education increased a lot during the past decade [4], but they are not frequently used at university level. [3] highlights that this is mainly due to psychological and social reason (e.g. entertainment and learning are often perceived as conflicting, especially at higher education level).

Nonetheless, there is evidence that serious gaming guarantees intrinsic motivation, generates challenging cognitive conflicts and provides situated learning based on learning concepts in accordance with constructivist psycho-cognitive theories [5].

Different types of games have already been employed for learning: board games, card games, role-playing games, first person shooter games, simulation games, management games, puzzle games, treasure hunts.

This study presents the design and implementation of a serious game for learning SQL called SQL Murder Mystery, inspired by the famous board game Cluedo [6]. The main contributions of the paper are listed as follow:

- We proposed a classification to distinguish the type of SQL queries based on their use in the game.
- We analyze team activity in the game to determine which behaviors most frequently influence success.
- We identified the advantages of using serious games in the context of our university course.

With respect to the last contribution, we answered the following research questions:

RQ1) Was the game helpful in understanding the students' level of knowledge of SQL? Does the success in the game actually correspond to good knowledge and familiarity with SQL language?

RQ2) Through the game can we identify knowledge gaps that are a high risk factor for success in the course?

II. Literature review

The use of board games has already been adopted in various educational contexts. In [7] the authors develop two serious games called Venture Creation Game and BattMan Simulator to review the process of developing a learning game and studying its effect in the learning experience of higher education students. The research group in [8] presents the Stereochemistry Game, a board/card game to help students in their studies of stereochemistry. [9] analyzes the use of a commercially available board game – The Logo Board Game – to facilitate student learning about branding elements. [10] describes the process of testing three modern board games (Steam, Codenames and Just One) to address the curricula of two civil engineering lectures.

[11] examines digital games for teaching software engineering, programming fundamentals, networks, algorithms complexity and security. The authors state that "digital games are still predominant, although there can also be observed a trend to non-digital ones (e.g. board games).".

Some other studies have adapted board games specifically for teaching programming courses. In [12] the authors add gaming elements in a virtual learning environment to teach an Algorithms and Programming Language course. The design includes several gamification components, such as: general details about the student profile, history of gained experience points, available and achieved challenges, leaderboard, redeemable points transfer and store. In [13] the authors present a case study to apply board games as a learning tool to improve Python programming language teaching. In [14] the authors investigate board games as a method to improve students' knowledge about the Java programming language.

978-1-6654-8810-5/22/$31.00 ©2022 IEEE
DOI 10.1109/COMPSAC54236.2022.00027

A final study we relied on for targeting our analyses is [15]; it remarks the potential of classifying the links between learning attributes and game mechanics as a means to scaffold teachers' understanding of how to perpetuate learning in optimal ways while enhancing the in-game learning experience.

III. Context of Research

A. University course

The context of our research is a university course on Databases, for students at the third year of a BS degree in Engineering. The number of enrolled students is 104. One of the main objective of the course is to learn the Structured Query Language (SQL) for querying databases, which covers about 50% of the program.

SQL is often perceived by students as a difficult language to learn, especially for those who have not a strong background on logic reasoning or computer programming [16]–[18]. An approach based on pedagogy of play [19], [20] can be helpful at any age because it enables experimentation and risk-taking, allows to explore ideas and involve emotions that can improve memorization [21]. Besides, games played in groups add a social component to learning, and this is the reason why the SQL game was proposed during a team-based learning session.

B. Team based learning

The game was played by 76 students and took place during a 4-hour lab session that used a team-based learning (TBL) approach ([22]–[24]). One of the major advantages of TBL is to allow learning in small groups inside large classroom, with a smaller number of expert facilitators (the two course instructors and one assistant in our specific context) with respect to project-based learning. The objective of the TBL session was to test the students' ability to perform SQL queries and their familiarity with the statements and clauses that will be assessed during the final exam. The complete list of learning goals (i.e. the SQL-related skills) is shown in TABLE I.

The steps involved in the TBL session were the following:

- *Pre-class preparation*: previous lectures, assignment and individual studies ensured a suitable preparation of the students for the TBL session.
- *Readiness Assessment Process (RAP)*: a multiple-choice quiz was proposed to the students and submitted first individually - *Individual Readiness Assurance Test (I-RAT)*, and then in groups - *Team Readiness Assurance Test (T-RAT)*. The proposed quiz was relative to the SQL learning goals and the time available to complete it was 30 minutes (15 minutes for I-RAT and 15 for T-RAT).
- *Application Oriented Activity*: it consists of an activity (AOA) to evaluate the team's problem solving skills relative to the learning goals. AOA should be significant, relevant, applicable and the same for all teams. Besides, the answer should be specific (correct or wrong) and must be submitted simultaneously by each team (simultaneous reporting). The time available to complete the task was 3 hours, and the proposed activity was the SQL game that will be discussed in the next subsection.
- *Class discussion*: in this phase teams can compare their AOA answers. It is essential for facilitators to focus discussion with thoughtful questions, to help students appropriately summarize their key points and support critical reflection. In our TBL session we had a discussion after about one hour and a half, when students were asked to submit a preliminary version of their answers and discuss them with the other teams. The second discussion was at the end of the game to compare teams' solution and provide feedback.

TABLE I: SQL learning goals

Id	**Description**
LG1	Select data from a database (`SELECT` statement)
LG2	Filter records based on conditions (`WHERE` clause)
LG3	Search for a specific text pattern in a column (`LIKE` operator + %)
LG4	Order records according to one or more columns (`LIMIT` and `ORDER BY` clauses + `ASC,DESC`)
LG5	Count table rows (`COUNT` aggregate function)
LG6	Use Inner Join: selects records that have semantic links between tables
LG7	Use Self-Join: join instances of the same table
LG8	Use Non Equi-Join: joins with comparison operators other than equal
LG9	Groups records into partitions (`GROUP BY`)
LG10	Apply aggregate functions (`SUM, MAX, MIN`, ...)
LG11	Return only different values (`DISTINCT` statement)
LG12	Return only the rows that always satisfy or never satisfy a condition (`NOT IN` operator)
LG13	Manage nested query (usually with `IN` and `NOT IN` operators)
LG14	Use correlation between queries

C. Game design

The game described in this study is called SQL Murder Mystery. The scenario is inspired by ”Cluedo” or ”Clue”, a very popular murder mystery board game that was invented in 1943 by the British board game designer Anthony E. Pratt. The game has already been redesigned for teaching SQL and it is available a Github repository licensed by MIT[1]. The original version of SQL Murder Mystery is designed to be both a self-paced educational tool to learn SQL concepts and commands and a funny game for experienced SQL users to solve an intriguing crime.

We made some changes to the original version to increase its complexity: in particular, more actors related to the murder have been added to the game and the database has been modified to increase the number of steps necessary to reach the solution, introducing more complexity in the additional steps to meet all the learning goals mentioned above.

At the beginning of the game, the following initial message is given to the players: *A crime has taken place and the detective needs your help. The detective gave you the crime*

[1] https://github.com/NUKnightLab/sql-mysteries

SQL Murder Mystery Schema

```
crime_scene_report(date, type, description, city)
person(id, name,  text, license_id, address_number, address_street_name, ssn)
drivers_license(id, age, height, eye_color, hair_color, gender, plate_number, car_make, car_model)
facebook_event_checkin(person_id, event_id, date, event_name)
get_fit_now_member(id, person_id, name, membership_start_date, membership_status)
get_fit_now_check_in(membership_id, check_in_date, check_in_time, check_out_time)
income(ssn, annual_income)
interview(person_id, transcript)
```

Fig. 1

scene report, but you somehow lost it. You vaguely remember that the crime was a murder that occurred sometime on Jan 15, 2018 and that it took place in SQL City. Start by retrieving the corresponding crime scene report from the police department's database.

Additionally, students receive credentials to connect to the SQL database. The database schema is shown in Figure 1.

Teams were asked to submit answers through a form consisting of a multiple choice question and an open-ended question:

1) How many people are connected to the murder? Consider only those roles (e.g. witness, murderer...) whose names can be retrieved.

   □ 2 □ 3 □ 4 □ 5 □ 6 □ 7

2) List all persons involved in the murder specifying their first name, last name, and role (e.g., murderer, witness...).

The first discussion phase, following a preliminary submission of the form after an hour and a half from the beginning of the game, focused on the first question: teams were asked to compare the number of people connected to the murder in various roles, but without mentioning names. The main facilitators' task was to ensure that the teams do not exchange more than this information. Besides, they encouraged teams to discuss the path they have taken so far, i.e., the tables they read that allowed them to identify some of the roles.

The main reason for this intermediate discussion was to avoid that teams stop their search believing they have found all the people connected to the murder, since they don't know the total number of *suspects*[2] at the beginning of the game. The goal of this phase is therefore to push the teams that have stopped their search to reflect based on the results given by other teams.

The steps and a possible solution of the game are shown in TABLE II. The steps modified from the original version of the game are underlined.

We can classify the steps of the game in two categories:

- *goals*: they correspond to the identification of a person related to the murder;
- *hints*: suggestions to reach to the next step of the game; in fact each identified person gives a deposition that serves as hint to get the next person.

In addition, the learning goals for which each step was designed are specified. The goals highlighted in bold indicate that they appear in the game for the first time, and that this step was expressly designated to meet that specific learning goal; the color (black or light blue) will be discussed in section V-E. In the following we will use the term *game goal* to indicate the discovery of a person related to the murder and the term *learning goal* to indicate the fulfilment of a specific SQL learning skill.

IV. Methodology

A. Data collection

The database ran on a server at our university. We defined some functions to store all query information (team, query text, query output, time). For each team we saved the data in a CSV file. We also recorded the answers submitted by the teams (through a Google form) in a CSV file.

B. Data processing

In this study, we are interested in understanding which game goals were achieved by each team but also to qualitatively analyze the student queries. We first determined when each team reached a specific game goal via a semi-automated process broken down into the following phases:

1) we use a script to run back the queries made by each team and to check if one of the suspects' names appears in the result table, with the only constraint that the number of rows is less than the threshold θ (set to 5) or that the name appeared in the first κ lines (set to 3);
2) we manually check the query logs to identify when a team reached a game goal with a less conventional and not automatically detectable path.

Then, we categorized the queries and identified which learning goals were fulfilled by each teams by solving the game goals.

1) Queries categorization: We manually determined the *key queries* for each team-goal pair, i.e., the set of queries used to reach that goal. These include only those strictly necessary to detect the *suspect*. In the ideal case shown in the proposed solution reported in TABLE II, one query is enough to solve

[2]The word *suspects* is used to identify all of the people connected to the murder, not necessarily those who may have committed the act (i.e. the first witness, the second witness, the killer, the instigator, the lover and the victim). Therefore, the identification of one suspect matches one game goal.

TABLE II: Game steps

Step	**Query**	**Output**	**Learning goals**

Starting Hint (HS)	`SELECT * FROM crime_scene_report`
	`WHERE type = "murder" AND city = "SQL City"`
	`AND date = "20180115";`
		Security footage shows that there were 2 witnesses. The first witness lives at the last house on ”Northwestern Dr”. The second witness, named Annabel, lives somewhere on ”Franklin Ave”.	LG1, LG2

Witness 1 (W1)	`SELECT id,name FROM person`
	`WHERE address_street_name = "Northwestern Dr"`
	`ORDER BY address_number DESC LIMIT 1;`
		14887, Morty Shapiro	LG1, LG2, **LG4**

Witness 2 (W2)	`SELECT id,name FROM person`
	`WHERE name LIKE "%Annabel%"`
	`AND address_street_name = "Franklin Ave";`
		16371, Annabel Miller	LG1, LG2, **LG3**

Hint from Witness 1 (HW1)	`SELECT transcript FROM interview` `WHERE person_id = "14887";`	I heard a gunshot and then saw a man run out. He had a ”Get Fit Now Gym” bag. The membership number on the bag started with ”48Z”. Only gold members have those bags. The man got into a car with a plate that included ”H42W”.	LG1, LG2
Hint from Witness 2 (HW2)	`SELECT transcript FROM interview` `WHERE person_id = "16371";`	I saw the murder happen, and I recognized the killer from my gym when I was working out last week on January the 9th.	LG1, LG2

Killer (K)	`SELECT p.id, p.name`
	`FROM person AS p, get_fit_now_check_in AS ci,`
	`get_fit_now_member AS m, drivers_license AS dl`
	`WHERE p.license_id = dl.id`
	`AND m.person_id = p.id`
	`AND ci.membership_id = m.id`
	`AND gender = ’male’`
	`AND membership_status = ’gold’`
	`AND plate_number LIKE ’%H42W%’`
	`AND membership_id LIKE ’%48Z%’`
	`AND check_in_date = ’20180109’`
		67318, Jeremy Bowers	LG1, LG2, LG3, **LG6**

Hint from Killer (HK)	`SELECT transcript FROM interview` `WHERE person_id = "67318";`	I was hired by the person who attended most events. She has always spoken to me while sitting in a Tesla Model S. Also, I know she has joined a gym although she never uses it because she doesn’t want to be seen.	LG1, LG2

Instigator (I)	`SELECT fe.person_id, p.name,`
	`COUNT(DISTINCT event_id) as n_events`
	`FROM facebook_event_checkin AS fe,`
	`get_fit_now_member AS gf,`
	`person as p, drivers_license AS dl`
	`WHERE fe.person_id = gf.person_id AND`
	`fe.person_id = p.id AND p.license_id = dl.id`
	`AND car_model = ’Model S’ AND car_make = ’Tesla’`
	`AND gf.id NOT IN (SELECT membership_id`
	`FROM get_fit_now_check_in)`
	`GROUP BY fe.person_id`
	`ORDER BY n_events DESC;`
		99716, Miranda Priestly	LG1, LG2, LG3, LG6, **LG9, LG10, LG11, LG12, LG13**

Hint from Instigator (HI)	`SELECT transcript FROM interview` `WHERE person_id = "99716";`	I can’t stand it when people lie. She always left the gym with her mistress.	LG1, LG2

Lover (L)
Victim (V)
Hint from Lover (HL)
	`SELECT p.name,p.id`
	`FROM person as p, get_fit_now_member as m`
	`WHERE p.id = m.person_id`
	`AND m.id IN (`
	`SELECT f1.membership_id`
	`FROM get_fit_now_check_in AS f1,`
	`get_fit_now_check_in AS f2`
	`WHERE f1.check_in_date = f2.check_in_date`
	`AND f1.check_out_time = f2. check_out_time`
	`AND f1.membership_id <> f2.membership_id`
	`GROUP BY f1.memebership_id`
	`HAVING COUNT(*) = (SELECT COUNT(*)`
	`FROM get_fit_now_check_in AS f3`
	`WHERE f3.membership_id = f1.membership_id)`
	`)`
		99991, Luisa Maicon
		99992, Kyra Slice
			LG1, LG2, LG3, **LG5**, LG6, **LG7, LG8**, LG9, **L613, LG14**

	`SELECT person_id,transcript FROM interview` `WHERE person_id = "99991" OR person_id = "99992";`	99991, I didn’t know she was engaged...I’m devastated, because I loved her...	LG1, LG2

each step of the game (i.e., the total number of key queries for each step could always be one); however, teams often preferred to perform multiple simpler queries.

Key queries are a subset of *focused queries*; given a specific game goal, *focused queries* include all queries in which at least one of the tables used in the *key queries* have been selected. They can be automatically determined. The idea is that a solution step can be reached by one *key query* or several *focused queries*, depending on the preferences of the students (complex or simple queries).

In a similar automated process we can distinguish two other types of queries: *exploratory queries* and *review queries*. The first ones correspond to queries in which all rows have been selected, to inspect table content. The second ones correspond to all queries in which the solutions of the previous steps have been reviewed, i.e. the key queries referred to the previous goals have been re-examined.

As an example, let's consider game goal K, i.e. the identification of the killer, and that team *t* provided the same solution reported in Table II. All queries starting from the one following the achievement of the previous game step will be considered. Those useful to review previous suggestions (hints) or solutions (goals) will be identified as *review queries* and those in which the content of a table is simply read as *exploratory queries*. Those in which at least one of the tables *person*,*get_fit_now_checkin*, *get_fit_now_member*, *drivers_license* appears will be tagged as *focused queries*. Some of them, the ones essential to find the killer will be additionally marked as *key queries*. Finally the remaining ones will be generically labeled as *other*, which identify the queries that select tables that are not useful in solving that specific game goal.

In case a query matches more than one type, the assignment follows this order: *exploratory*, *review*, *focused*.

2) Setting learning goals: For each solution submitted by a teams for a specific game step, we annotated which *learning goals* were satisfied. This was helpful to understand the degree to which the team solution differs from the designed one in terms of reached *learning goals* .

V. Game analysis

This section presents the main findings of this work. The analysis is divided into the following subsections: *Background analysis* focuses on the analysis of the results obtained in the initial *RAT* tests. *Game goals analysis* is centered on the game goals achieved by each team. *Game step analysis* analyzes the different paths taken by teams to solve the mystery. *Query style analysis* examines queries qualitatively, evaluating the impact of the query style (*review, exploratory, focused*) on the achieved results. *Learning goals coherence* focuses on the coherence between learning goals and game goals, to understand if success in the game corresponds to actually improving SQL skills. *Risk factors identification* shows the main SQL knowledge gaps revealed by the game and Students' feedback reports the students' level of satisfaction about the game experience. Finally *Discussion* summarizes the answers to the research questions.

A. Background analysis

TABLE III: I-RAT SQL results

Team	Mean value	Median value	Max value	Min value
Team 1	6.69	7.00	7.50	5.50
Team 2	7.00	7.00	8.00	6.00
Team 3	6.56	6.50	10.00	4.50
Team 4	7.67	8.00	9.50	5.00
Team 5	6.17	6.25	8.00	4.50
Team 6	6.00	6.00	9.00	4.00
Team 7	7.25	7.25	9.00	5.50
Team 8	6.19	6.25	7.00	5.00
Team 9	6.50	7.00	8.50	3.50

TABLE III shows the individual student scores on the initial I-RAT test; the maximum, minimum , mean, and median values are reported for each team. Only one student achieved the maximum score (10 points out of 10). To infer whether the difference between teams is significant, we calculated both Paired Student's t-test and Wilcoxon Signed Rank Test for each pair of teams; the p-values obtained indicate that we cannot reject the null hypothesis for all pairs of teams hence there is not a significant difference between the scores means of two groups. Regarding the T-RAT test, all teams achieved the maximum score of 10.

We can therefore conclude that both considering the groups members individually and the groups as a whole, the teams have similar SQL background knowledge.

B. Game goals analysis

Figure 2 shows the game goals achieved by each team. All teams found at least one person connected to the murder. Team 8 matched the fewer people, i.e. the first witness and the murderer. Five out of 9 teams reached 3 game goals: all of them found the two witnesses, and moreover Team 1, Team 2, Team 4 and Team 6 found the murderer, while Team 7 found the instigator. Team 3 and Team 5 identified the two witnesses, the killer and the instigator. Only Team 9 fully solved the game. As a result, the majority of teams achieved at least half of the game goals.

C. Game step analysis

The sequence of steps in the solution proposed in TABLE II is HS , W1 , W2 , HW1 , HW2 , K , HK , I , HI , L , V , HL . This order is not the only possible to solve the mystery, and it may differ in some sub-sequences:

- W1 , W2 , HW1 , HW2 can be replaced by one of the following options: 1) W2 , W1 , HW2 , HW1 2) W2 , W1 , HW1 , HW2 3) W1 , HW1 , W2 , HW2 4) W2 , HW2 , W1 , HW1 . In fact, witnesses are identified before other people because they give hints to find the following suspects but the first one can be identified before the second one and vice versa because the initial advice HS is helpful for finding both.

- L, V and HL can be reached simultaneously because the lover suggestion HL is helpful to understand the lover and the victim names. L gives suggestions for solving both steps.

Given these constraints, most teams mostly maintained the order of the designed steps. Only Team 2 and Team 8 followed different paths. Team 2 identified the instigator without having found the first witness, the killer, and their hints. In fact, the team members understood from the starting hint and the witness hints that the murder was related to a gym. Consequently they made the following query using the interview table:

```
SELECT transcript
FROM  interview I
WHERE transcript LIKE '%gym%' LIMIT 0, 25
```

As a result, Team 2 was able to smartly solve I, with less effort in terms of SQL queries complexity; however, the different path did not allow them to identify the witness and the killer and they stalled after having found the instigator.

Team 8, on the other hand, was able to directly identify the killer without the second witness hint. Using the first witness suggestions they were able to limit the possible people to a small subset and then they verified which of them had an associated interview. Again, the different path allowed them to solve the specific game goal but not to proceed with the following ones.

D. Query style analysis

We analyzed the teams' behavior based on the categories of queries discussed in IV-B1.

Figure 3 shows the percentage of queries of each category for each game goal and overall. In addition, the categorization related to the initial suggestion HS is shown. This hint is more worthy of investigation than the others because the latter are always found at the same time or shortly after the corresponding game goal (few queries to analyze). The graph shows that *exploratory* percentage is greater at this stage than in the following ones: it is common for teams to navigate in the database to inspect the tables and understand the attributes especially when they are not yet familiar with the schema. On the contrary, we do not notice game phases where *review* and *focused* are significantly more prevalent than others.

Looking at the overall bars and at Figure 4, that shows the relationship between query categories and total number of game goals, the main finding is related to Team 9: it is the only team that completed the game and it shows the highest percentage of *focused* queries, very few *exploratory* and *review* query and no queries at all labeled as *other*. This suggests that they were very focused and proceeded straight forward to the solution . The same does not apply to other teams as most of them achieved the same number of game goals with different percentage of *focused* and *other* queries.

Our data then are coherent with the hypothesis that success in the game is more likely to happen when the team is highly goal-oriented (very high *focused* query percentage and really few *other* queries)).

E. Learning goals coherence

The goal of this subsection is to assess whether the experiment met the learning design expectations. We start focusing on analyzing what learning goals each team met in their solutions.

For each goal considered, we distinguish between *specific* and *related* learning goals. The first ones are those specifically designed to solve that game goal and they are usually related to complex SQL clauses: the second type of goals are relative to simpler SQL statements and, even if they can contribute to solve the considered step, they have not been specifically designed for that. Looking back at TABLE II, *specific* learning goals are black and highlighted in bold, while *related* learning goals are colored in light blue. In addition we labeled as *divergent* learning goals those met by the teams without being related to the learning goals for that exercise; they are not present in TABLE II.

Figure 5 shows the results for each game goal and overall. Colored squares indicate that the learning goal was met by the team; light blue means that the learning goal is *specific*, while black suggests that the learning goal is *related* and red means that the learning goal is *divergent*. The squares with the black and white grid show that the team did not solve the corresponding game goal. The overall picture does not show the "color" of the learning goal but only its eventual fulfilment.

For the first two game goals (**W1,W2**) the teams have proposed solutions similar to the designed one. Team 4 found the most complex solutions, fulfilling also some learning goals *specific* of the following steps. In the first 3 game goals (**W1,W2,K**) at least half of the teams fulfilled the *specific* learning goals.

For the last steps of the game fewer groups were able to reach the solution and consequently the learning goals were met by a few teams; however, 2 out of the 4 teams that solved **I** also met all *specific* game goals.

Considering the overall results, only 2 teams (3, 9) met more than half of the learning goals at least once in their proposed solutions; to figure out if these are the same teams that overcame more obstacles in the game, we compared the learning goals met with the game goals satisfied. In particular we focus on adherence between learning goals and game goals, to understand if success in the game corresponds to actual SQL skills acquisition (see Figure 5).

Team 9 met most learning goals and it is also the only team that finished the game. Team 3, which succeeded in 4 out of 6 game goals, also showed confidence with 10 out of 7 learning goals. Team 4 although having accomplished more learning goals than teams 1, 2, 6 and 7 reached the same performance in the game. As we previously pointed out, this is because it met more learning goals than the designed ones for the first steps. Team 8, which is the one that got stuck earlier (only two game goals reached) also achieved just two learning goals.

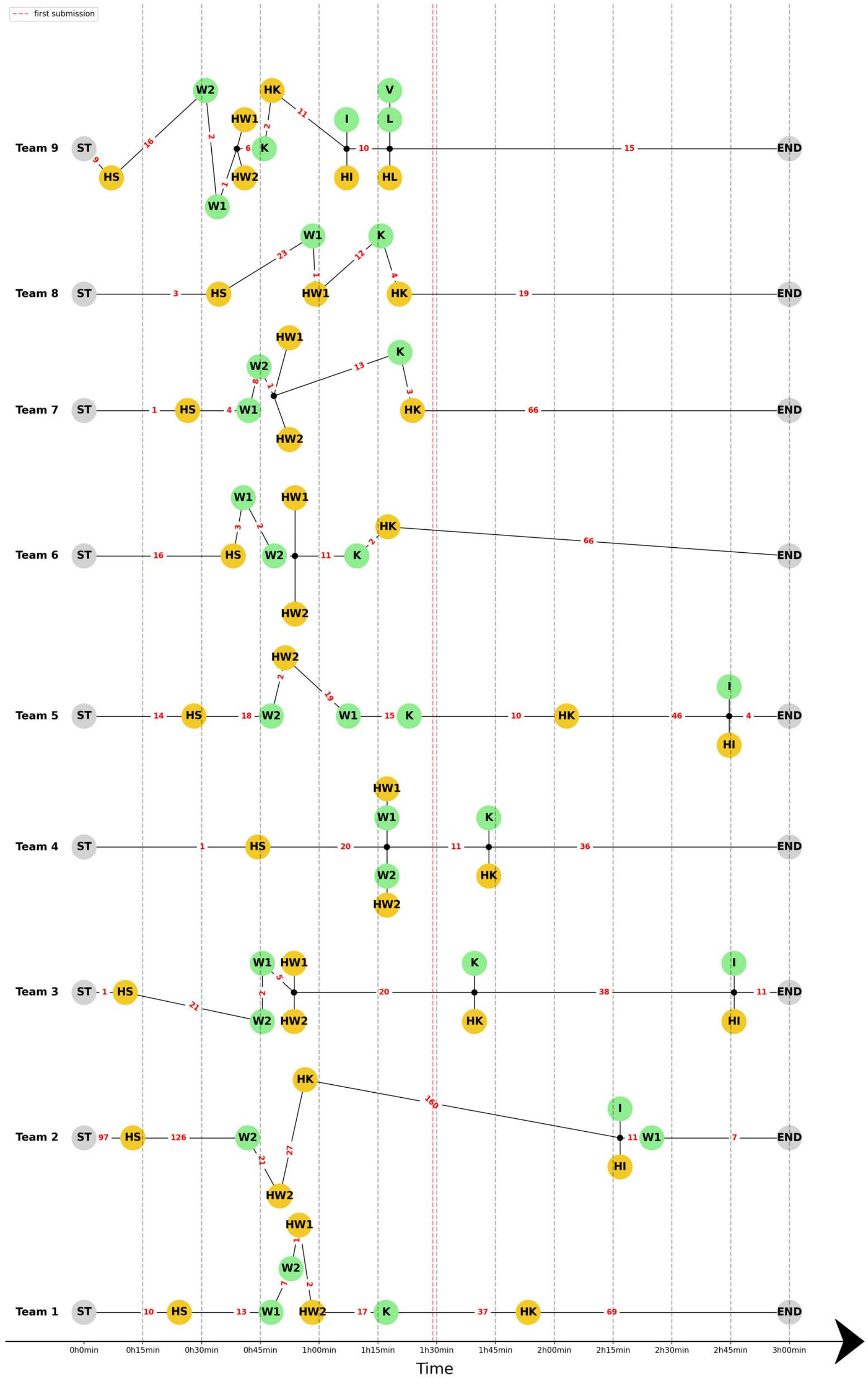

Fig. 2: Teams' step order

The figure shows the teams paths, i.e. when each game step has been reached; steps are represented by the circles: the green ones indicate *goals* and the yellow ones indicate *hints*. The smaller black circles are used when a team reaches two game steps simultaneously. Note that the vertical displacement of the circles has not a specific meaning, but it is used to avoid overlaps in the drawing. Finally, the red numbers indicate the total number of queries made between two game steps.

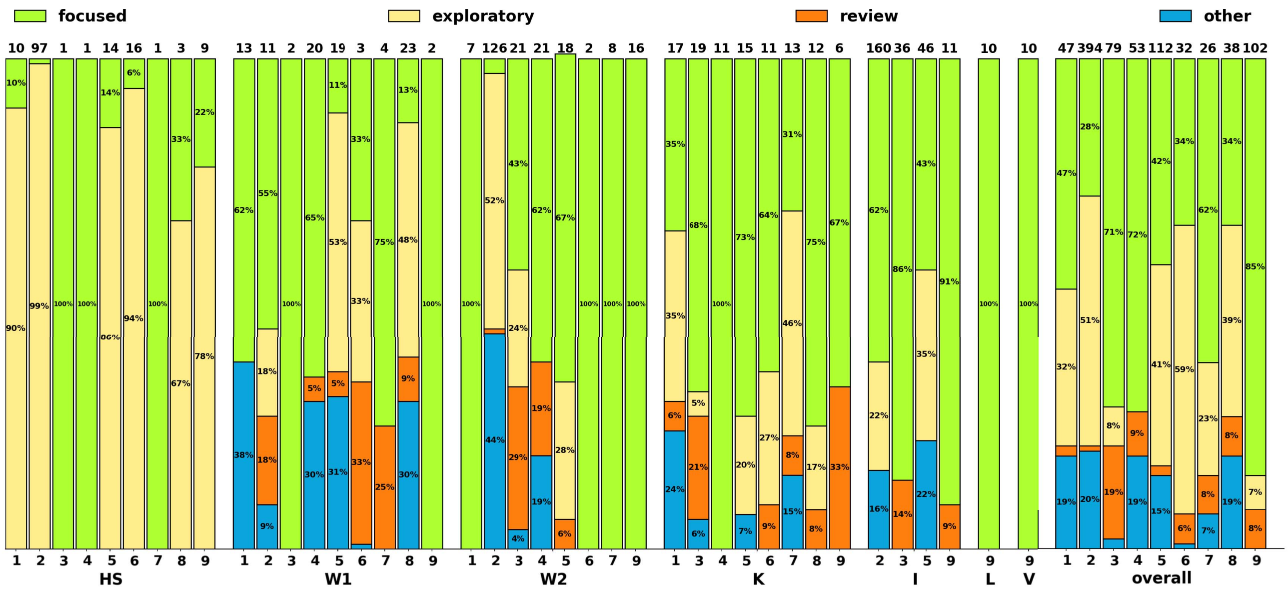

Fig. 3: Query categories

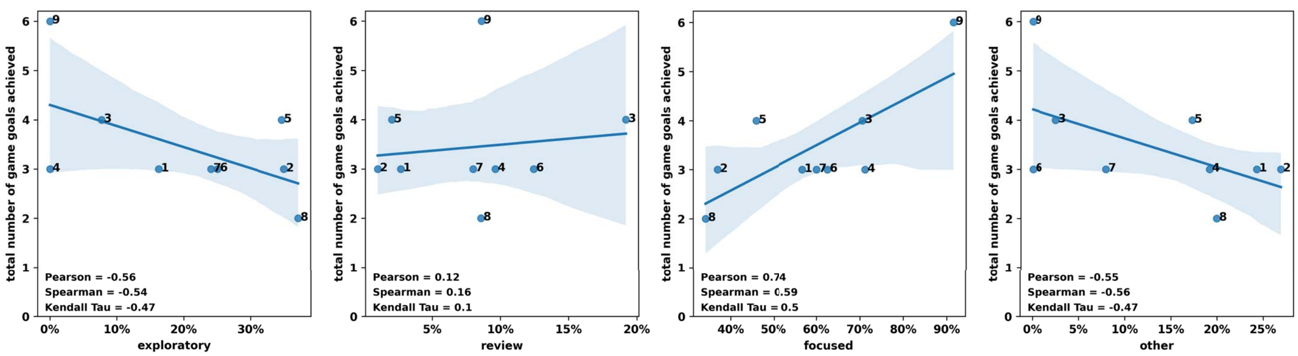

Fig. 4: Query categories with respect to game goals

The correlation coefficients show a fairly positive correlation between the total number of learning goals and the total number of game goals. In particular, Kendall Tau which is the most reliable metric with little non-continuous ordinal data, takes on the highest value than in all previous figures.

Figure 6 shows the number of queries useful to reach the game goals; the darker color of the bars indicates that a greater number of learning goals have been satisfied. The solution paths that adopted a greater number of queries are those for which fewer learning goals were addressed. This figure is also helpful in explaining the behavior of team 5, which is an outlier: it submitted only simple queries thus satisfying fewer learning goals but reached more game goals than most teams. However, the more complex goals (**K,I**) required more queries with respect to the other teams.

TABLE IV

Team	Last step	Next goal	Reason for failure	
			SQL-related	Game-related
1	HK	I	`DISTINCT` missing (LG11)	
2	HK,HI	K,I	`DISTINCT` missing (LG4) Sort missing (LG11)	
3	HI	V,L		The instigator hint was not followed
4	HK	I	`DISTINCT` missing (LG11)	
5	HI	V,L		End of time
6	HK	I	`DISTINCT` missing (LG11)	
7	HI	V,L	`DISTINCT` missing (LG11)	
8	HK	I	`DISTINCT` missing (LG11)	

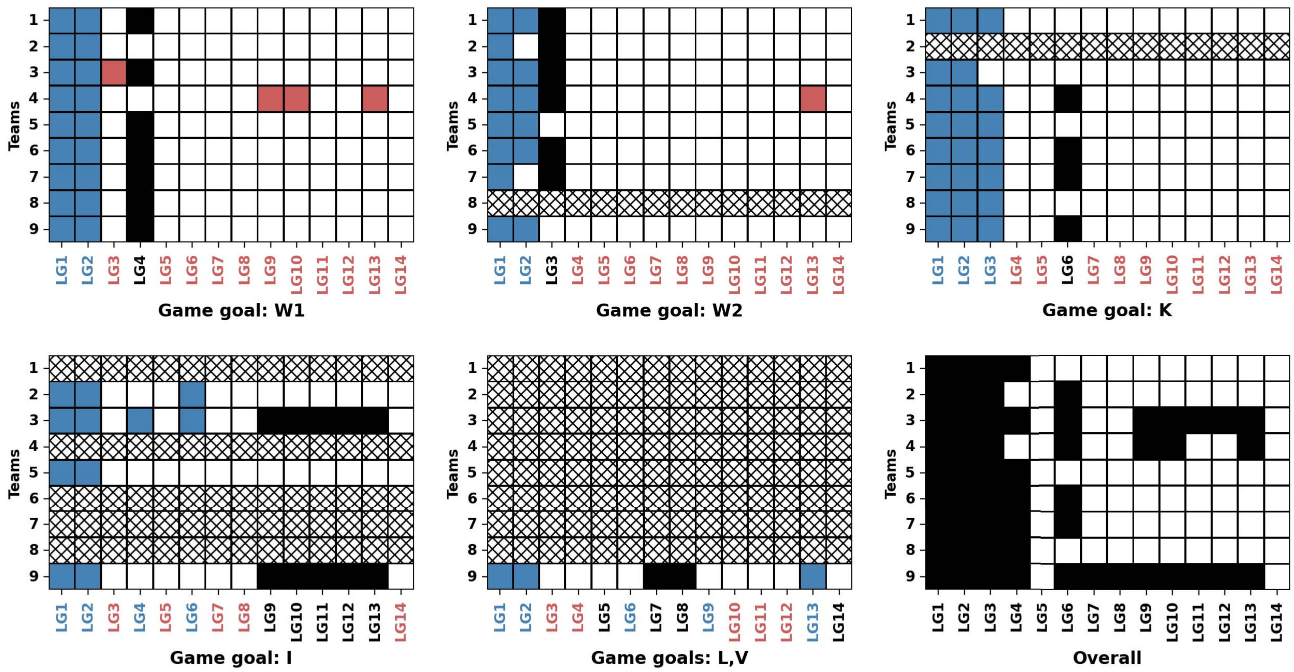

Fig. 5: Achieved learning goals

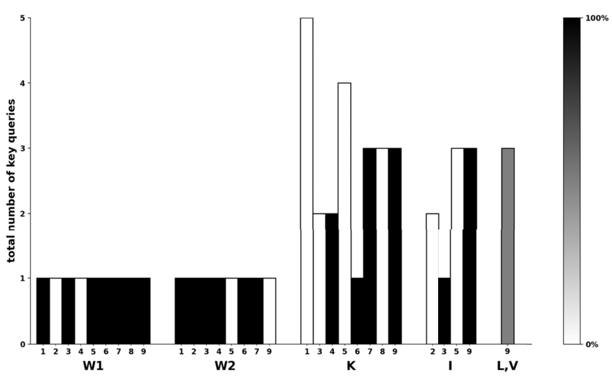

Fig. 6: Adherence to specific learning goals

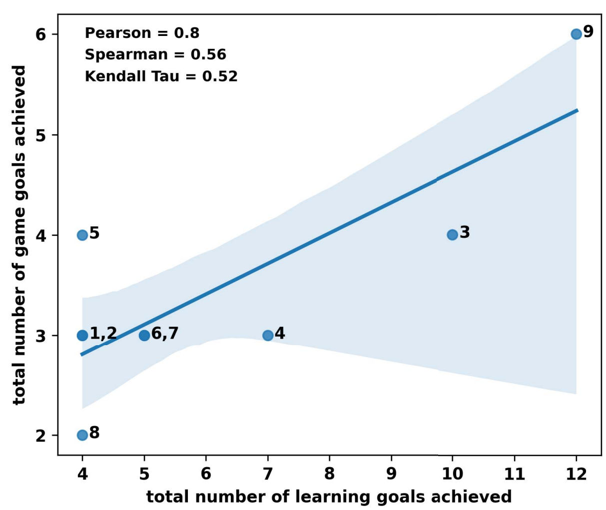

Fig. 7: Learning goals vs game goals

F. Risk factors identification

We examined the query logs to understand the major barriers faced by teams that failed to complete the game. They are reported in Table IV.

The majority of teams (5 out of 9) failed to move forward after getting the hint from the killer. The main reason for the failure is due to a knowledge gap regarding the SQL `DISTINCT` statement.

A sentence from **HK** states: *I was hired by the person who attended most events.*. This sentence should clarify or at least introduce a doubt that the events a person attended more than once should not be counted multiple times.

The teams which got stuck for `DISTINCT` counted 38 students; indeed, it was not a good sign that none of them thought to use the `DISTINCT` statement. This turned out to be unexpected for the instructors since during the lectures special attention was paid to the different uses of `DISTINCT`, by showing similar examples. This finding was very useful because it revealed a major knowledge gap that was recovered in the following teaching lectures.

G. Students' feedback

Finally, we can report that during the last lecture of the course, the instructor asked the student to fill a qualitative questionnaire regarding the course satisfaction, where two questions where about the TBL session experience. Among the 48 students that answered, 20 rated it as excellent, 21 as good and 7 as fair. The comments were generally very positive, and the adjectives used by the students were: challenging, funny, novel, interesting, gratifying, useful for assessing one's knowledge. The only complain was that teams were too large, suggesting groups of no more than 4 people.

H. Discussion

In the following we answer to the research questions.

RSQ1) Was the game helpful in understanding the students' level of knowledge of SQL? Does the success in the game actually correspond to good knowledge and familiarity with SQL language?: Based on the analysis reported in subsection V-E, the game design was actually useful in assessing students' familiarity with SQL and the teams that showed higher skill in SQL in most cases are the ones that met more game goals.

RSQ2) Through the game can we identify knowledge gaps that are a high risk factor for success in the course?: The game was helpful in identifying the main educational gap of the students, that it is linked to **LG11**, which measures students' confidence with the `DISTINCT` statement. The identification of the risk factor allows the instructor to react and fill the knowledge gap in the following lectures.

VI. Conclusion and Future Work

In this paper we presented the design of a game called SQL Murder Mystery designed for testing SQL skills. The game was played during a team-based learning 4-hour lab session with students attending a university course on databases.

The implementation of the game matched the design expectations, as the teams that achieved the best results revealed good fulfilment of the SQL language learning goals. In addition, we analyzed the activity of teams in playing the game, revealing the most useful behaviors to succeed and the main knowledge gap of the players.

Future work will focus on:

- repeat the experiment to collect more data in order to infer whether the findings of this study are valid only for the laboratory session under consideration or can be generalized;
- introducing small variations to the game in order to understand which steps are the most significant.

References

[1] D. Djaouti, J. Alvarez, and J.-P. Jessel, "Classifying serious games: the g/p/s model," *Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches*, 01 2011.

[2] B. Gross, "Handbook of research on serious games for educational applications," *Editorial Advisory Board*, pp. 402–405, 2016.

[3] A. Darwesh, "Concepts of serious game in education," *International Journal Of Engineering And Computer Science*, 01 2016.

[4] D. Liberona and C. Rojas, "Entrepreneurship knowledge transfer through a serious games platform," in *International Conference on Knowledge Management in Organizations*. Springer, 2017, pp. 144–156.

[5] H. Mouaheb, A. Fahli, M. Moussetad, and S. Eljamali, "The serious game: What educational benefits?" *Procedia - Social and Behavioral Sciences*, vol. 46, pp. 5502–5508, 2012, 4th WORLD CONFERENCE ON EDUCATIONAL SCIENCES (WCES-2012) 02-05 February 2012 Barcelona, Spain. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S187704281202201X

[6] H. van Ditmarsch and B. Kooi, "Cluedo," in *One Hundred Prisoners and a Light Bulb*. Cham: Springer International Publishing, 2015, pp. 109–121. [Online]. Available: http://link.springer.com/10.1007/978-3-319-16694-0_11

[7] D. Liberona, S. Ahn, M. Lohiniva, P. Garate, and C. Rojas, "Serious games usage in higher education, experiences and guidelines," in *Learning Technology for Education Challenges*, L. Uden and D. Liberona, Eds. Cham: Springer International Publishing, 2021, pp. 138–150.

[8] J. N. D. Junior, D. Uchoa, M. Lima, and A. Monteiro, "Stereochemistry game: Creating and playing a fun board game to engage students in reviewing stereochemistry concepts," *Journal of Chemical Education*, vol. 96, 06 2019.

[9] A. Khan and G. Pearce, "A study into the effects of a board game on flow in undergraduate business students," *The International Journal of Management Education*, vol. 13, no. 3, pp. 193–201, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S147281171500021X

[10] M. Sousa, "Modern serious board games: modding games to teach and train civil engineering students," in *2020 IEEE Global Engineering Education Conference (EDUCON)*, 2020, pp. 197–201.

[11] P. Eduardo Battistella and C. Gresse von Wangenheim, "Games for teaching computing in higher education – a systematic review," *IEEE Technology and Engineering Education*, vol. 1, pp. 8–30, 03 2016.

[12] A. C. T. Klock, A. N. Ogawa, I. Gasparini, and M. S. Pimenta, "Integration of learning analytics techniques and gamification: An experimental study," in *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)*, 2018, pp. 133–137.

[13] D. B. Jordaan, "Board games in the computer science class to improve students' knowledge of the python programming language," in *2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC)*, 2018, pp. 1–5.

[14] ——, "Board games in the computer science class to improve students' knowledge of the java programming language: A lecturer's perspective," in *Proceedings of the 2nd International Conference on Education and Multimedia Technology*, ser. ICEMT 2018. New York, NY, USA: Association for Computing Machinery, 2018, p. 1–4. [Online]. Available: https://doi.org/10.1145/3206129.3239425

[15] P. Lameras, S. Arnab, I. Dunwell, C. D. Stewart, S. Clarke, and P. Petridis, "Essential features of serious games design in higher education: Linking learning attributes to game mechanics," *Br. J. Educ. Technol.*, vol. 48, pp. 972–994, 2017.

[16] A. Ahadi, J. Prior, V. Behbood, and R. Lister, "A quantitative study of the relative difficulty for novices of writing seven different types of SQL queries," in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, Jun. 2015.

[17] J. B. Smelcer, "User errors in database query composition," *Int. J. Hum. Comput. Stud.*, vol. 42, no. 4, pp. 353–381, Apr. 1995.

[18] S. Brass and C. Goldberg, "Semantic errors in SQL queries: A quite complete list," *J. Syst. Softw.*, vol. 79, no. 5, pp. 630–644, May 2006.

[19] R. Farné, "Pedagogy of play," *Topoi*, vol. 24, pp. 169–181, 09 2005.

[20] J. Mcgonigal, *Reality is broken: Why games make us better and how they can change the world*. Penguin Press, 2011.

[21] C. M. Tyng, H. U. Amin, M. N. M. Saad, and A. S. Malik, "The influences of emotion on learning and memory," *Front. Psychol.*, vol. 8, p. 1454, Aug. 2017.

[22] L. K. Michaelsen, A. B. Knight, and L. D. Fink, Eds., *Team-based learning*. Sterling, VA: Stylus Publishing, Feb. 2004.

[23] A. Burgess, C. van Diggele, C. Roberts, and C. Mellis, "Team-based learning: design, facilitation and participation," *BMC Medical Education*, vol. 20, no. S2, p. 461, Dec. 2020. [Online]. Available: https://bmcmededuc.biomedcentral.com/articles/10.1186/s12909-020-02287-y

[24] H. P. Whitley, E. Bell, M. Eng, D. G. Fuentes, K. L. Helms, E. D. Maki, and D. Vyas, "Practical Team-Based Learning from Planning to Implementation," *American Journal of Pharmaceutical Education*, vol. 79, no. 10, p. 149, Dec. 2015. [Online]. Available: http://www.ajpe.org/doi/10.5688/ajpe7910149