



Learning-outcome-based curriculum redesign and student acceptance: evaluation of the relational database course in systems engineering

Ernesto Galvis-Lista¹

Alexander Bustamante-Martinez²

^{1,2}Universidad del Magdalena, Santa Marta, Colombia.

¹Email: egalvis@unimagdalena.edu.co

²Email: abustamante@unimagdalena.edu.co



(✉ Corresponding Author)

Abstract

This article evaluates the impact of redesigning the Relational Databases course in the Systems Engineering program at Universidad del Magdalena. In 2022, the course was reformulated to promote higher-order thinking by coupling an outcomes-based design with active, participatory pedagogy. The new version was offered over six consecutive semesters (2022-1 to 2024-2); across these cohorts, 317 students passed the course, and this study analyzes 107 complete surveys from students who both passed and consented to respond. Results indicate strong acceptance: 97% rated the experience "Excellent" or "Good," and 90% reported that the outcomes-based approach improved learning relative to content-centered courses. Moreover, 86% felt they learned "better" or "much better" than in other courses. A k-means analysis of performance revealed three profiles: 57% with high, homogeneous achievement; 21% particularly strong in relational design (Learning Outcome 3); and 22% with intermediate attainment. Despite these gains, approximately two-thirds of respondents overestimated their own level, pointing to a metacognitive challenge worth addressing in future iterations. Open-ended responses highlight the instructor's role, content quality, and applied practice as key strengths. Suggestions emphasize increasing project-based work and formative feedback. Overall, the findings support the value of an outcome-oriented redesign for fostering higher-order learning in database education and motivate targeted strategies, especially feedback and self-assessment, to calibrate student judgments and consolidate learning.

Keywords: Active learning, Engineering education, Learning assessment, Learning-outcome-based education, Relational databases, Student perception.

Citation | Galvis-Lista, E., & Bustamante-Martinez, A. (2026). Learning-outcome-based curriculum redesign and student acceptance: evaluation of the relational database course in systems engineering. *Journal of Education and E-Learning Research*, 13(2), 241–250. 10.20448/jeelr.v13i2.8810

History:

Received: 6 August 2025

Revised: 5 May 2026

Accepted: 27 May 2026

Published: 15 June 2026

Licensed: This work is licensed under a [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/)

Publisher: Asian Online Journal Publishing Group

Funding: This study was conducted with internal resources provided by Universidad del Magdalena. No external funding was received.

Institutional Review Board Statement: The Institutional Ethics Committee (CEI) of Universidad del Magdalena (Santa Marta, Colombia) reviewed this activity on September 29, 2025, confirmed adherence to ethical principles and the use of informed consent, and noted that the submission occurred post-activity; it recommended filing a prospective protocol for future iterations. All analyzed data were anonymized to protect participant confidentiality.

Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

Competing Interests: The authors declare that they have no competing interests.

Authors' Contributions: Both authors contributed equally to the conception and design of the study. Both authors have read and agreed to the published version of the manuscript.

Contents

1. Introduction	242
2. Review of Literature.....	244
3. Methods	245
4. Results	246
5. Conclusions	248
References.....	249

Contribution of this paper to the literature

This manuscript provides an outcomes-based redesign of an undergraduate Relational Databases course across six cohorts (2022–2024), coupling student evidence (n=107) with k-means performance segmentation to derive three learner profiles. It uniquely documents a metacognitive overestimation effect and distills transferable design implications, project-based work, formative feedback, and structured self-assessment, advancing evidence-based curriculum improvement in database education.

1. Introduction

Data are a strategic asset for organizations because sound management supports both day-to-day operations and decision-making (Valles-Coral, 2023). In this context, relational databases have become a widely adopted solution in companies of all sizes. As a result, undergraduate computing programs typically include at least one course devoted to the design, construction, and use of relational databases (Salleh, Lim, & Chua, 2024).

Several pedagogical approaches exist for the first database course. These range from traditional models, centered on theoretical exposition and predominantly lecture-based instruction, to active-learning approaches in which students play the central role in their own learning process. Table 1 summarizes some of these approaches (Ishaq et al., 2023).

Table 1. Approaches to Teaching the Introductory Database Course.

Approach	Description
Theory-based approach (Traditional)	Lecture-centered classes that emphasize theory and formalism. Assessment relies mainly on written exams, with limited in-class practice.
Project-based learning (PBL)	The course is organized around an applied project (Design and implementation of a real database). The project usually carries significant weight in the final grade.
Active and collaborative learning	Strategies that engage students in tasks during class—solving exercises, discussing brief case studies, and working cooperatively in groups.
Flipped classroom and blended learning	A hybrid mode in which students study foundational content in advance (e.g., through videos or online readings), while face-to-face sessions focus on hands-on activities and clarifying doubts.

Developing the skills needed to design, build, and use databases, like other computing competencies, is a significant challenge because it is a creative process that follows no exact formulas (Hamzah, Rukun, Rizal, Purwati, & Zarnelly, 2019). Multiple studies have identified recurring obstacles in teaching relational databases, chief among them: (a) an over-emphasis on theory, which produces students with basic conceptual knowledge but limited applied skills; (b) the persistence of traditional, lecture-based models that, despite multimedia enhancements, overload students with content and dampen their motivation; and (c) practical activities that are too brief and simplistic, far removed from real-world contexts, thereby limiting the effective application of knowledge (Hamzah et al., 2019).

Within the university setting, courses are expected not merely to develop instrumental skills but also to foster higher-order cognitive abilities (Yanti & Thohir, 2024). In this regard, the revised Krathwohl, Bloom, and Masia (1964) offer a useful framework for engineering education, establishing levels that move beyond simple knowledge application, remember, understand, apply, analyze, evaluate, and create (Britto & Usman, 2015). These levels help structure learning experiences that promote critical thinking, problem-solving, and innovation.

This vision contrasts with approaches focused exclusively on transmitting technical content and has been widely adopted to guide curriculum design, instructional planning, and learning assessment in engineering programs, particularly in courses such as relational databases. To illustrate the application of Bloom's taxonomy to relational databases, Table 2 is presented. Each row corresponds to one level, providing a description and an example related to the use of Structured Query Language (SQL).

Table 2. Illustration of the expected performance at each outcome level.

Level	Description of the level	Example applied to SQL SELECT
Remember	Retrieve and recall basic information or previously learned concepts without requiring deep understanding.	State the basic syntax of a SELECT query.
Understand	Demonstrate meaningful comprehension of concepts through explanation, comparison, or description.	Explain the difference between the WHERE and HAVING clauses in a SELECT query.
Apply	Use learned information to solve problems or perform tasks in new contexts or practical situations.	Write a query that shows the identifier column and the concatenated first- and last-name columns for all records in the table of employees where the salary value exceeds 5,000.
Analyze	Break information into parts to understand underlying relationships, patterns, and structures.	Analyze the result sets of two similar queries and determine the conditions that cause the observed discrepancies
Evaluate	Make judgments based on specific criteria or standards to validate solutions or decisions taken.	Evaluate whether a proposed SELECT query for the monthly sales report meets the defined requirements for accuracy and performance.
Create	Generate or construct something new by integrating learned and original elements to solve complex problems	Build a query that produces a ranking of the five best-selling products by region.

Accordingly, the curriculum design of the Relational Databases course at Universidad del Magdalena (Santa Marta, Colombia) targets three learning outcomes aligned with the upper levels of the revised Bloom's taxonomy. These outcomes encompass the use, design, and construction of relational databases and are specified as follows.

1. Build SQL scripts that meet data-retrieval needs in a transactional relational database.
2. Design, at the conceptual and logical levels, relational databases that satisfy structured-data storage requirements for a specific organizational context.
3. Implement the objects defined in the logical design, together with other elements required for relational-database operations, in a database-management system widely used in industry.

These outcomes foster key professional competencies: querying data effectively to address specific needs, designing database models that align with organizational requirements, and implementing objects that enhance data querying and manipulation.

To determine each student's level of achievement, an ordinal five-category scale was employed, ranging from Basic to Exceptional, each linked to a quantitative band (60%–100%). This quantitative reference allows conversion between scales, for example, representing the Basic level as 300 on a 0–500 scale.

For each category, a performance descriptor was defined, specifying the evidence expected from the student. The descriptor uses language consistent with, and traceable to, the wording of the learning outcome, while detailing aspects related to complexity.

Consequently, the performance descriptors for Learning Outcome 1 are presented in Table 3. They show progression in SQL query complexity, from simple queries with basic functions and single-table filtering (Basic) to complex, multi-table queries that include execution plan analysis (Exceptional).

Table 3. Performance descriptors by achievement level for Learning Outcome 1.

Learning Outcome 1 (LO1)	
Build SQL scripts that meet data-retrieval needs in a transactional relational database	
Level	Descriptor
Basic (60%)	Build SQL scripts that retrieve data stored in a single table, using diverse filtering criteria, scalar and aggregate functions, and row grouping.
Intermediate (70%)	Build SQL scripts that retrieve data stored in one or several tables by means of standalone subqueries and table joins.
Advanced (80%)	Build SQL scripts that retrieve data from multiple tables employing correlated subqueries and set operations.
Outstanding (90%)	Build SQL scripts that retrieve aggregated or summarized data according to various criteria, calculated with window functions.
Exceptional (100%)	Build SQL scripts that satisfy complex, multi-table data-retrieval needs using operations that achieve acceptable performance, as confirmed by the DBMS execution-plan analysis.

Likewise, Table 4 presents the performance descriptors for Learning Outcome 2, and Table 5 for Learning Outcome 3. This structure ensures consistency in assessment and traceability throughout the course's progressive development.

Table 4. Performance descriptors by achievement level for Learning Outcome 2.

Learning Outcome 2 (LO2)	
Design, at the conceptual and logical levels, relational databases that meet structured-data storage requirements for transactional data identified within a specific organizational context.	
Level	Descriptor
Basic (60%)	Designs a conceptual-level database that represents all elements required to satisfy the specified needs, using standard entity–relationship notation, entities, relationships, and attributes, to ensure a clear and effective structure.
Intermediate (70%)	Designs all necessary elements at the logical level to satisfy the identified needs; however, improvements are needed, particularly in the appropriate selection of data types for each column domain and in the implementation of column-level or multi-column constraints.
Advanced (80%)	Produces a logical-level design that complies with normalization rules up to at least the third normal form (3NF) and applies recognized design patterns and best practices
Outstanding (90%)	Demonstrates the effectiveness of the design by creating the principal tables of the logical model, on paper or digitally, and populating them with demonstration data that reflect anticipated scenarios, thereby validating the design's functionality and accuracy.
Exceptional (100%)	Evaluates the quality of the designs created, ensuring full compliance with normalization rules, established patterns, and best practices.

Table 5. Performance descriptors by achievement level for Learning Outcome 3.

Learning Outcome 3 (LO3)	
Implement the objects specified in the logical design, along with other objects required for operating a relational database, in a DBMS that enjoys wide industrial use.	
Level	Descriptor
Basic (60%)	Creates all tables specified in the logical design by using SQL or a graphical tool provided by the DBMS.
Intermediate (70%)	Creates the SQL scripts required to insert reference or demonstration data into the database tables.
Advanced (80%)	Adds column-level constraints and indexes that improve query execution performance.
Outstanding (90%)	Creates scalar functions and stored procedures (parameterized queries).
Exceptional (100%)	Implements triggers that handle insert, update, or delete events on the database tables.

The course adopts a mixed strategy that combines several of the approaches listed in Table 1. Face-to-face sessions concentrate on schema analysis, troubleshooting, and the structured review of designs and scripts. Students work both individually and in teams, tackling challenges that simulate real-world scenarios in the

development and operation of relational databases. Teamwork facilitates the detection of conceptual errors, the discussion of modeling alternatives, and evidence-based decision-making. The course also promotes reflection on organizational and ethical aspects of data management, such as faithful representation of reality, protection of sensitive data, and traceability. The instructor acts as a facilitator, guiding the process from a critical, holistic perspective.

Course sequence. The course begins with activities that involve intensive use of SQL on existing relational databases, allowing students to familiarize themselves with real structures and develop querying skills across various domains (e.g., commerce, sports, leisure). Building on this experience, database design is introduced to meet specific needs framed as contexts or user stories. Modeling is addressed iteratively, progressing from simple structures to complete relational schemas that incorporate relationships, constraints, and hierarchies. In the final stage, students develop objects such as views, indexes, functions, procedures, and triggers to integrate them into the solution of authentic problems.

Capstone project. A final collaborative project challenges students to design and build a database solution that demonstrates attainment of the three learning outcomes. The project is contextualized within a specific domain; in one academic term, for example, the domain encompassed three world-class sporting competitions: Formula 1, the Vuelta a España, and the Roland-Garros tennis tournament. A set of functional and technical requirements serves as terms of reference, simulating the conditions of a professional contract in industry. Figure 1 summarizes the course's main characteristics, methodology, pedagogical focus, collaborative work, didactic sequence, instructor role, and organizational context.

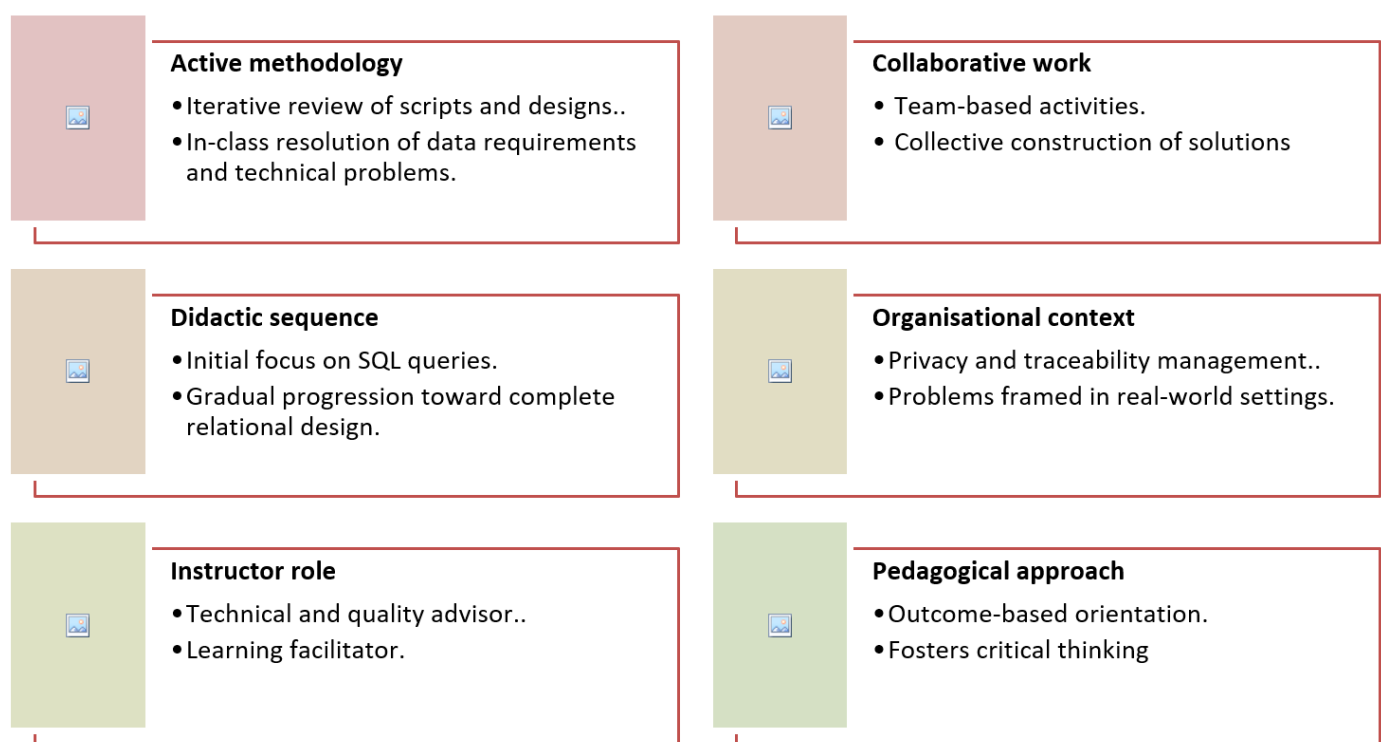


Figure 1. Relational database course main characteristics.

The general approach of the Relational Databases course outlined above was first implemented in the initial academic semester of 2022. After six consecutive semesters, an evaluation was undertaken to gauge the perceptions of students who completed and passed the course. This exploratory, descriptive study is guided by the research question: What is the perceived quality of the redesigned course, and how strongly is it associated with learning outcome (LO) attainment? The findings are presented in this article; the following sections detail the methodology employed, the results obtained, and their analysis, and conclude with lessons learned from the process.

2. Review of Literature

The introductory database course is a cornerstone in the education of computing professionals and therefore occupies a central place in most curricula. Its prominence has stimulated a substantial body of research spanning teaching and assessment methods, supporting tools, and curricular guidelines. Traditional approaches, typically a theory-heavy lecture followed by a lab session, have been questioned for their effectiveness in sustaining students' interest and motivation (Al-Dmour, 2010; Ishaq et al., 2023). This has motivated a first line of inquiry focused on active, student-centered pedagogies, with particular attention to Project-Based Learning (PBL). Consistent with this diagnosis, Freeman et al. (2014) reported a meta-analysis of 225 STEM studies showing that active-learning sections increased examination performance by about 6 % and reduced failure rates by a factor of 1.5 relative to traditional lecturing. In the specific case of SQL, a cornerstone topic in database courses, the systematic mapping by Taipalus and Seppänen (2020) identifies teaching and assessment methods, learning tools, and error diagnosis as the dominant research streams, while signaling a persistent scarcity of studies on advanced SQL, replication, and longitudinal effects.

PBL has been adopted to teach “learning to design by designing”: student teams plan, build, and present an authentic database solution while using tools to manage the activities involved (Domínguez & Jaime, 2010; Tomas, Iserte, & Perez, 2021; Tomás, Pérez Francisco, & Marqués Andrés, 2019). Studies report that this approach not only increases engagement but also strengthens the integration of conceptual design, logical modeling, and implementation skills, competencies that are essential for professional practice. Recent meta-analytic evidence reinforces these findings: Hidayat et al. (2024) synthesised two decades of PBL implementations in engineering education and reported a significant positive effect on higher-order thinking skills, particularly for implementations

of 9 to 18 weeks with small student groups. Beyond pedagogy, Miedema, Taipalus, and Aivaloglou (2023) showed that the application domain chosen for database assignments, sports, music, or socially relevant scenarios—systematically influences novice engagement, reinforcing the case for contextualised, authentic projects rather than abstract toy schemas.

Aligned with this goal, research has also examined alternative pedagogical strategies for teaching complex topics such as SQL and database design. Notably, studies report that gamification has a positive effect on learning SQL, with most primary investigations finding increased student motivation and improved academic performance (Del-Pozo-Arcos & Balderas, 2024; Langlands & Morales-Trujillo, 2023). However, gamification can also elicit anxiety or impatience, so benefits are not guaranteed. Its effectiveness hinges on calibrating both the duration and the difficulty of activities. Evidence further suggests that students in gamified settings commit fewer semantic errors, while competitive variants are particularly effective at reducing syntax errors. Flipped classroom models also feature prominently in literature, with studies reporting significant gains in student performance. Flipped designs have shown positive effects on withdrawal/pass rates and on the distribution of grades: in a replicable self-evaluation of an introductory course, the failure rate decreased while the share of top grades increased, suggesting that “flipping” the fundamentals offers a scalable path to guided self-study (Meng, 2021; Welk, 2025). Additional evidence indicates that the flipped mode can stimulate interest, strengthen autonomous learning, and deepen students’ understanding and mastery of course content. Beyond pedagogy, a complementary line of research addresses curriculum design itself. Constructive alignment, proposed by Biggs (1996), argues that deep learning is achieved when intended learning outcomes, teaching activities, and assessment tasks are coherently linked, so that students “construct” meaning by engaging in activities that directly rehearse the targeted outcome. In higher-education implementations, operationalized Biggs’s model and demonstrated that a backward-designed, aligned curriculum measurably facilitates the attainment of intended outcomes, provided that assessment criteria remain traceable to outcome descriptors. This principle is the cornerstone of the redesign examined in the present study, in which each learning outcome is associated with a rubric of progressive performance descriptors (Tables 3–5).

Complementing pedagogical shifts, researchers have explored tools that improve usability and deliver immediate feedback. LearnER, developed through a collaboration between the University of South-Eastern Norway (USN) and Kristiania University College (KUC), has been deployed across six courses over four academic years to teach core data-modeling skills. Students solve assignments of varying difficulty and receive automated, often visual, feedback; gamified elements (e.g., point deductions for help requests and leaderboards) encourage extensive practice (Dæhli, Kristoffersen, Lauvås, & Sandnes, 2021). In a similar vein, a tool by Stefanidis and Koloniari (2016) helps students grasp decomposition techniques and apply them to achieve Third Normal Form or Boyce–Codd Normal Form; survey results show more favorable student perceptions than with conventional classroom instruction. For SQL, visualization-oriented systems such as viSQLizer have been used to scaffold learning through interactive representations of queries and results (Folland, 2016). Finally, a concern that directly motivates the present study is the calibration of students’ self-perceptions against objective attainment. Writing in this journal, found that learning activities explicitly structured along the revised Bloom’s taxonomy significantly improved students’ metacognitive abilities, their capacity to monitor what they know and how well they are performing. Along similar lines, also in this journal, reported that systematic formative-assessment techniques measurably strengthen students’ self-assessment skills. Together, these findings frame the metacognitive gap documented later in our results (Section 4) and justify the feedback-centered actions proposed in the conclusions.

3. Methods

To explore student perceptions of the new curricular approach and identify opportunities for improvement, a four-stage methodology was defined. The study followed an exploratory and descriptive design based on primary data. The stages are summarized below.

Stage 1: Delimitation of the target population. All students from Universidad del Magdalena who passed the Relational Databases course between semesters 2022-1 and 2024-2 were identified, resulting in a universe of 317 individuals. Institutional records were cleansed to ensure only students with a definitive passing grade were included.

Stage 2: Questionnaire design. An instrument was created to capture students’ self-assessment of (1) how much they learned, (2) how they value the importance of what they learned, and (3) their overall impression of the course. The conceptual design resulted in an online questionnaire with 13 questions organized into three sections. Table 6 lists the questions in each section along with the response options.

Stage 3: Data collection. An invitation to complete the questionnaire was emailed to all 317 students who had passed the course between the first semester of 2022 and the second semester of 2024. The link was sent to institutional addresses, accompanied by an initial message and three staggered reminders up to April 2025. A total of 107 fully completed responses were received. The resulting CSV file was cleaned by verifying integrity, removing duplicates, and standardizing values. Respondents were distributed as follows: six from semester 2022-I, 15 from 2022-II, 15 from 2023-I, 23 from 2023-II, 26 from 2024-I, and 22 from 2024-II.

Table 6. Questions and response options for the applied instrument.

Section 1: Achievement of course learning outcomes		
#	Question	Response options
1	One expected outcome of the Databases course focuses on querying data with SQL to answer relevant needs or questions within the specific organizational domain in which the database exists. Select all statements that describe what you are able to do after passing the course.	The descriptors of the achievement levels for the first learning outcome (Table 3)
2	Another expected outcome of the Databases course focuses on the conceptual and logical design of relational databases that meet structured data storage needs for transactional data in a specific organizational context. Select all statements that describe what you	The descriptors of the achievement levels for the second learning outcome (Table 4).

Section 1: Achievement of course learning outcomes		
#	Question	Response options
	are able to do after passing the course.	
3	The final expected outcome of the Databases course focuses on implementing the objects specified in the logical design, along with other objects necessary for operating a relational database, using an industry-standard DBMS. Select all statements that describe what you are able to do after passing the course.	The descriptors of the achievement levels for the third learning outcome (Table 5)
4	In general, the level of learning you achieved in the Databases course was:	Basic, Intermediate, Advanced, Outstanding, Exceptional
5	Compared with the learning you achieved in other courses in your degree program, the level you achieved in the Databases course is:	Much worse, Worse, The same, Better, Much better
Section 2: Importance of the learning achieved in the course		
#	Question	Response options
6	Rate the usefulness of what you learned in the Databases course for taking and passing subsequent courses in the curriculum.	Not useful, Slightly useful, Moderately useful, Very useful, Extremely useful
7	Rate the importance of what you learned in the Databases course regarding its applicability to your professional life.	Not important, Slightly important, Moderately important, Very important, Extremely important
Section 3: Course design and development		
#	Question	Response options
8	Select all statements that, in your opinion, describe aspects or characteristics of the Databases course you took:	Multiple choice on statements that describe the course
9	Compared with content-centered courses, the learning-outcomes focus of the Databases course makes the learning process:	Much worse, Worse, The same, Better, Much better
10	Overall, the Databases course you took and passed is:	Poor, Fair, Acceptable, Good, Excellent
11	What did you like most about the Databases course?	Open text
12	What did you like least about the Databases course?	Open text
13	What do you suggest to improve the Databases course?	Open text

Stage 4 focused on data analysis. The responses to the closed-ended questions were processed using descriptive statistics (frequencies, percentages, measures of central tendency), whereas the open-ended answers were coded through a thematic analysis approach in four phases: (1) familiarization and initial hybrid coding, deductive codes aligned with course components (project-based tasks, flipped sessions, immediate-feedback tools) and inductive codes for emergent ideas; (2) codebook refinement on a pilot subset; (3) double-coding of the full corpus by two trained researchers, estimating inter-rater agreement and resolving discrepancies by consensus; and (4) consolidation of themes with explicit inclusion/exclusion criteria. To substantiate qualitative claims, we report theme prevalence (percent of respondents mentioning each theme). The results were organized into tables and charts to facilitate interpretation and discussion.

4. Results

Table 7 presents the results obtained from responses to the first three questions, which focus on the level of achievement students believe they reached for each of the three expected learning outcomes. One row is provided for each learning outcome, showing the number of students who placed their performance level in the corresponding category. Each qualitative performance level is associated with a quantitative value expressed as a percentage.

Table 7. Self-perceived level of achievement attained in the course.

Learning outcome	Performance levels				
	Basic (60%)	Intermediate (70%)	Advanced (80%)	Outstanding (90%)	Exceptional (100%)
LO1. Build SQL scripts that meet data-retrieval needs in a transactional relational database	1	5	16	24	61
LO2. Design, at the conceptual and logical levels, relational databases that meet structured-data storage requirements for transactional data identified within a specific organizational context.	1	2	9	9	86
LO3. Implements the objects specified in the logical design, along with other objects required for operating a relational database, in a DBMS that enjoys wide industrial use.	6	6	22	16	57

The table shows the distribution of achievement levels, with the “Exceptional” category predominating, chosen by most students in all three outcomes (61 in LO1, 86 in LO2, and 57 in LO3), followed by “Outstanding.” The “Basic” and “Intermediate” levels are in the minority. LO2 receives the highest rating, with 80% (86/107) in “Exceptional,” whereas LO3 exhibits the greatest relative dispersion (ranging from 6 in “Basic” to 57 in “Exceptional”). This table demonstrates that students largely attribute high or outstanding achievement levels to themselves across all three outcomes, with a particularly elevated perception for LO2.

Using the performance percentages, a clustering algorithm was applied. Figure 2 shows the elbow method output for determining the optimal number of clusters in a k-means analysis. The within-cluster sum of squares drops sharply from k=1 to k=3, after which the slope flattens, indicating that three clusters strike a balance between internal compactness and simplicity.

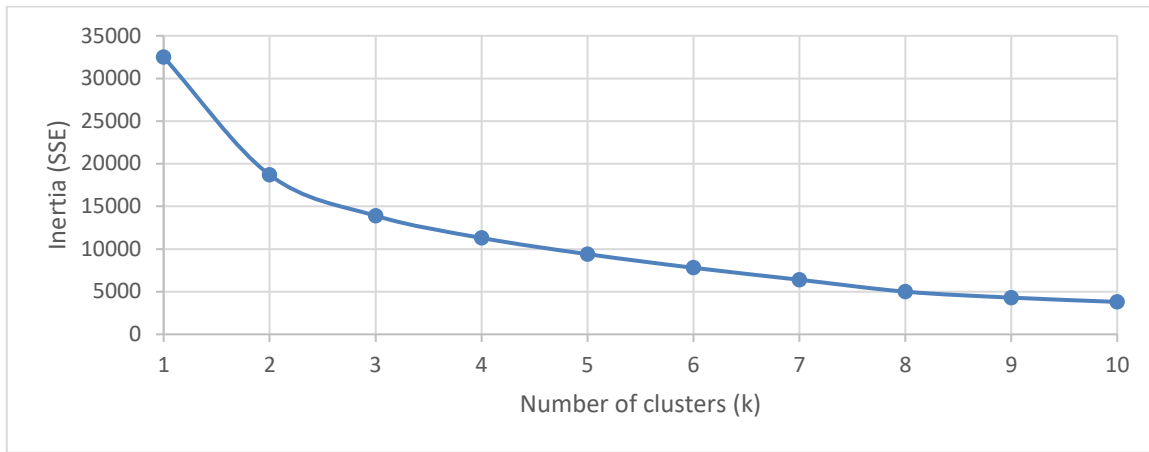


Figure 2. Application of the elbow method.

The application of the k-means algorithm with three clusters ($k = 3$) revealed the three profiles listed in Table 8. Figure 3 graphically compares their performance across the three LOs: Cluster 1 ($n = 62$) exhibits high and homogeneous achievement (means ≥ 96 , low dispersion); Cluster 2 ($n = 15$) shows medium-to-low and highly variable performance, with its weak point in LO2 (mean 79.3); Cluster 3 ($n = 30$) excels in LO2 (mean 99) but drops markedly in LO3 (mean 76).

Table 8. Cluster characteristics (means, dispersion, and ranges) by LO.

Cluster	Count	RA1				RA2				RA3			
		Mean	Std.	Min.	Max.	Mean	Std.	Min.	Max.	Mean	Std.	Min.	Max.
1	62	96.4	6.08	80	100	99.5	2.16	90	100	98.35	3.70	90	100
2	15	84.0	14.04	60	100	79.3	7.98	60	90	86.66	13.45	60	100
3	30	90.3	9.27	70	100	99.0	3.05	90	100	76.00	8.13	60	90

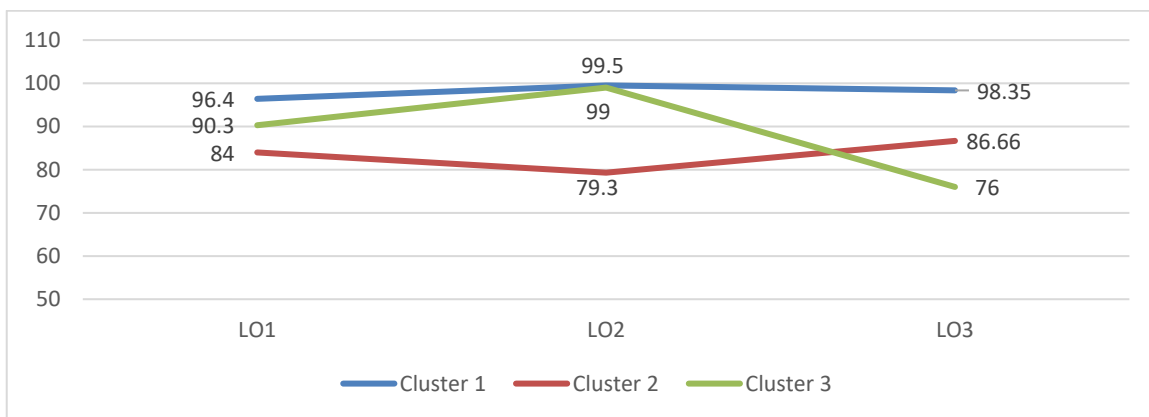


Figure 3. Performance Profile by Group.

In the comparison between the grade obtained and the declared self-assessment, most students, as shown in Table 9, overestimate their performance: 72 out of 107 ($\approx 67\%$) place themselves one level higher than warranted by their final mark. A second group, 26 students ($\approx 24\%$), matches the target level exactly, whereas only 9 ($\approx 8\%$) underestimate their achievement. This pattern points to a significant metacognitive gap: two out of every three students perceive themselves as having reached a higher mastery than their results demonstrate, highlighting the need for feedback strategies that help align self-perception with reality and reinforce more realistic evaluation criteria.

Table 9. Comparison between the final grade obtained and the overall perceived performance.

		Self-assessment					Total
		Basic	Intermediate	Advanced	Outstanding	Exceptional	
Grade	Basic	5	9	6	7	2	29
	Intermediate	2	5	14	9	1	31
	Advanced	0	3	1	16	3	23
	Outstanding	0	0	4	11	5	20
	Exceptional	0	0	0	0	4	4
	Total	7	17	25	43	15	107

The comparative perception of the course is clearly positive: 86% of students rate it as “Better” (37%) or “Much better” (49%) than other courses in their program, whereas only 10% regard it as “The same” and 4% as “Worse.” In short, the course stands out against most others in students’ views, although a few dissatisfied cases should be examined to further improve the learning experience. Additionally, Figure 4 shows the course’s perceived usefulness for subsequent subjects in the curriculum, and Figure 5 illustrates how important students consider it for their professional careers.

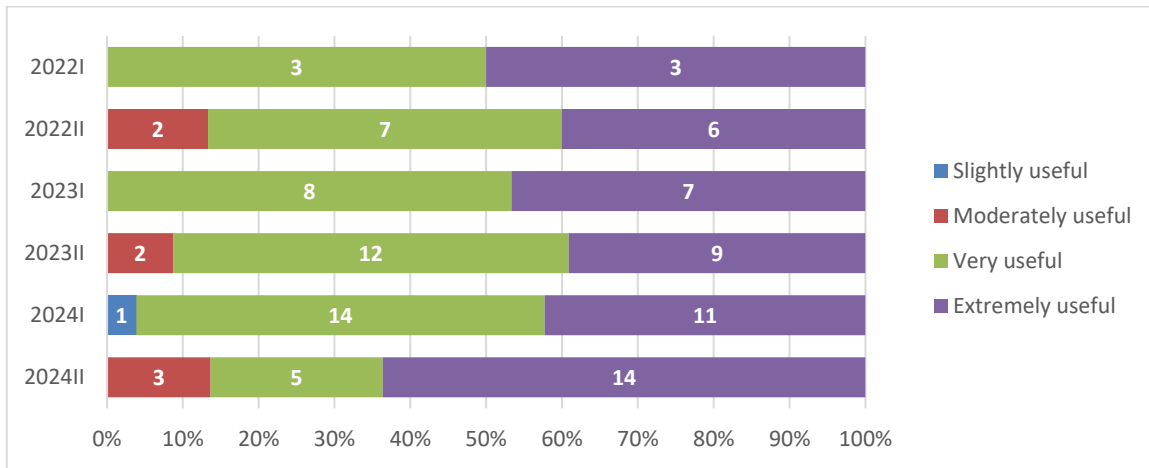


Figure 4. Perception of the course's usefulness for taking and passing subsequent courses.

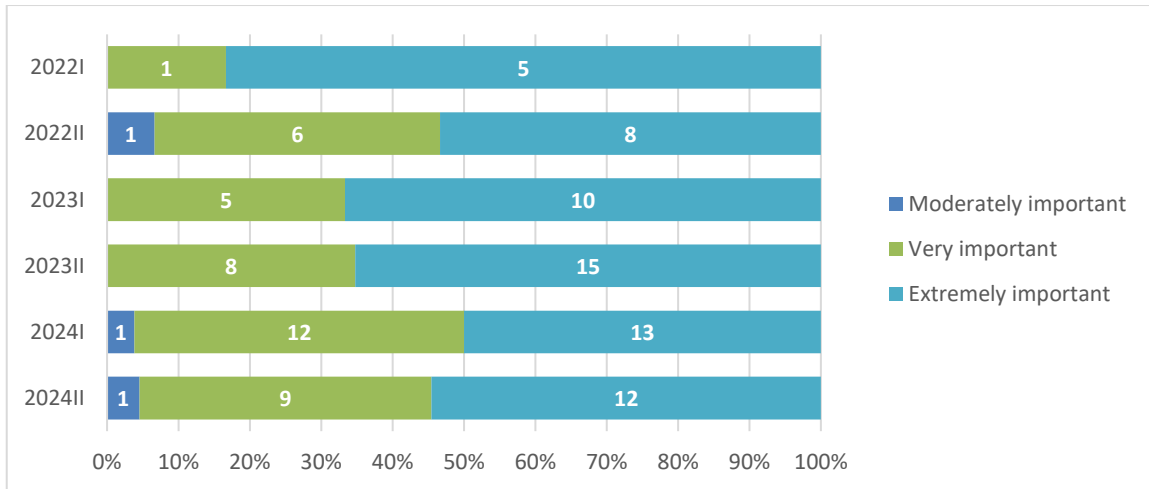


Figure 5. Perception of the importance of what was learned in the course for professional life.

Students also clearly underscore its pedagogical value. Ninety percent state that the learning-outcomes focus makes the learning process “Better” or “Much better” than content-centered courses, and, consistently, 97% rate the course overall as “Excellent” (77%) or “Good” (21%). The concordance between these two metrics suggests that structuring the course around learning outcomes translates into a highly satisfactory experience, perceived as superior to traditional courses.

Regarding the open-ended questions about the course’s strong points, the arguments align with the characteristics presented in Figure 1. Table 10 sets out the qualitative categories extracted from the open-response data.

Table 10. Student-highlighted aspects of the course.

Topics	Keyword examples	Frequency (n)
Teacher / Explanations	Teacher, classes, explanations, dynamics	62
Content / Materials	Topics, content, data, materials	40
Methodology	Methodology, outcomes, structure, approach	27
Practice / Laboratory	Practice, lab, exercises, workshops	14
Projects	Project, projects	14

It is observed that the most frequently cited category is “Teacher / Explanations” (n = 62), which connects directly with the “Instructor role,” as students identify the lecturer as both a technical guide and a learning facilitator. The second theme, “Content / Materials” (n = 40), aligns with the “Didactic sequence” laid out for the course; beginning with SQL queries and advancing toward a full relational design was perceived as a clear and well-structured progression. Emphasis on “Methodology” (n = 27) and “Practice / Laboratory” (n = 14) reflects the effectiveness of the “Active methodology,” centered on solving real-world problems. Finally, the appreciation of “Projects” (n = 14) supports both the component of “Collaborative work,” by fostering the collective construction of solutions, and the “Organizational context,” by situating activities in scenarios that demand a level of performance similar to professional practice. Taken together, these findings confirm the coherence between the course design principles and the learning experience.

Regarding aspects to improve or strengthen, analysis of responses to what you like least about the course and what you suggest improving the course reveals that two-thirds of students did not highlight any significant negatives. Among criticisms, two main points stand out: certain instrumental aspects of assessment (17%) and class methodology/dynamics (17%), with comments about a fast pace. To a lesser extent, observations concern dense content (10%) and a perceived heavy workload (9%).

5. Conclusions

The course achieved a high level of outcomes: an average final grade of 78% translated into 90% of students reaching “Advanced” or higher in at least two of the three learning outcomes (LO1–LO3). A k-means analysis (k = 3) identified a majority group (57%) with homogeneous, outstanding performance; a second group (21%) strong in LO3 but moderate in LO1–LO2; and a third group (22%) with medium-to-low performance. Comparing objective grades with self-assessment revealed a moderate correlation ($\rho \approx 0.58$) but showed that 67% overestimate their

level, confirming the need to strengthen metacognition. In addition, 86% perceive that this course surpasses others in their program, and 90% believe that the outcome-centered approach improves learning compared with content-centered courses. To deepen the interpretation of these findings, future analyses should incorporate student-profile covariates—semester standing, prior academic performance, course repeat status, gender, and prior participation in project-based activities. Including these variables will help disentangle cohort effects from prior preparation, reveal subgroup heterogeneity, and improve the precision of estimates. They will also enable targeted tests of metacognitive calibration, examining whether the accuracy of students' self-evaluations systematically varies with prior achievement or experiential background. Such disaggregation would sharpen practical recommendations and inform equity-aware course refinements.

In 107 open-ended responses, students highlighted five positive aspects: the instructor's role and clarity of explanations ($n = 62$), content quality (40), active methodology based on outcomes (27), laboratory practice (14), and applied projects (14). By contrast, the main negative observations focused on assessment/feedback (18) and methodological dynamics (18), although two-thirds did not report unfavorable aspects. Suggested improvements emphasize expanding authentic projects (23 suggestions), refining feedback and rubrics (10), and increasing guided practice sessions (9). These findings support the coherence between the instructional design, rooted in Bloom's taxonomy, progressive sequences, and real-world problem solving, and the student experience, while pointing to specific adjustments.

To deepen and consolidate these results, we propose three complementary actions: first, to implement a formative-feedback plan supported by analytic rubrics and immediate feedback, allowing us to gauge its impact on students' self-perception and performance; second, to integrate learning analytics, drawing on platform logs, script-execution records, and project participation, to model study profiles and design personalized interventions; and third, to replicate the outcome-based, active-learning approach in other core courses, comparing control and experimental cohorts through metrics of achievement, perception, and transfer of professional competencies. Together, these actions will refine the pedagogical strategy and yield robust evidence of its long-term effectiveness.

Building on these results, we envision four lines of inquiry. First, conduct longitudinal tracking into advanced courses and capstone/thesis modalities that require applying the database learning outcomes, linking course-level attainment to downstream academic performance. Second, implement graduate tracer studies to assess real-world impact, e.g., role alignment, time-to-placement, supervisor evaluations, and task performance, thereby estimating the course's contribution to workplace effectiveness. Third, incorporate direct assessment of soft skills (teamwork, communication, data ethics) using validated rubrics and situational judgment tasks embedded in project work. Fourth, deepen the study of self-regulation and metacognition by designing diagnostic instruments and calibration metrics that compare students' self-ratings with objective evidence of attainment, and by testing light-touch interventions (prompted reflection, feedback nudges) to reduce miscalibration. Finally, we will replicate the methodology and instructional adjustments in adjacent foundational courses, programming, software engineering, and advanced databases, to evaluate transfer effects on engagement and demonstrable learning outcomes.

References

- Al-Dmour, A. (2010). A cognitive apprenticeship-based approach to teaching relational database analysis and design. *Journal of Information & Computational Science*, 7(12), 2495-2502.
- Biggs, J. (1996). Enhancing teaching through constructive alignment. *Higher Education*, 32(3), 347-364. <https://doi.org/10.1007/BF00138871>
- Britto, R., & Usman, M. (2015). *Bloom's taxonomy in software engineering education: A systematic mapping study*. Paper presented at the 2015 IEEE Frontiers in Education Conference (FIE), IEEE.
- Dæhli, O., Kristoffersen, B., Lauvås, J. P., & Sandnes, T. (2021). Exploring feedback and gamification in a data modeling learning tool. *Electronic Journal of e-Learning*, 19(6), 559-574. <https://doi.org/10.34190/ejel.19.6.2501>
- Del-Pozo-Arcos, S., & Balderas, A. (2024). Gamifying SQL learning: Strategies, impact, and trends in higher education. *Interaction Design and Architecture(s)*(63), 156-182. <https://doi.org/10.55612/s-5002-063-008>
- Domínguez, C., & Jaime, A. (2010). Database design learning: A project-based approach organized through a course management system. *Computers & Education*, 55(3), 1312-1320. <https://doi.org/10.1016/j.compedu.2010.06.001>
- Folland, K. A. T. (2016). *viSQLizer: Using visualization for learning SQL*. Paper presented at the Norwegian ICT Conference for Research and Education: 2016 Norwegian Informatics Conference.
- Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23), 8410-8415. <https://doi.org/10.1073/pnas.1319030111>
- Hamzah, M. L., Rukun, K., Rizal, F., Purwati, A. A., Hamzah, & Zarnelly. (2019). A review of increasing teaching and learning database subjects in computer science. *Revista Espacios*, 40(26), 6-15.
- Hidayat, H., Anwar, M., Harmanto, D., Dewi, F. K., Orji, C. T., & Mohd Isa, M. R. (2024). Two decades of project-based learning in engineering education: A 21-year meta-analysis. *TEM Journal*, 13(4), 3514-3525. <https://doi.org/10.18421/TEM134-84>
- Ishaq, M., Abid, A., Farooq, M. S., Manzoor, M. F., Farooq, U., Abid, K., & Helou, M. A. (2023). Advances in database systems education: Methods, tools, curricula, and way forward. *Education and Information Technologies*, 28(3), 2681-2725. <https://doi.org/10.1007/s10639-022-11293-0>
- Krathwohl, D. R., Bloom, B. S., & Masia, B. B. (1964). *Taxonomy of educational objectives: The classification of educational goals, Handbook II: Affective domain* (A. Anderson, Ed.). New York: David McKay.
- Langlands, E., & Morales-Trujillo, M. (2023). *Learning from errors: An empirical study on the impact of gamification on SQL query formulation*. Paper presented at the Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education.
- Meng, B. (2021). *Practice of flipped classroom teaching mode of computer course-taking access database as an example*. Paper presented at the 2021 4th International Conference on Information Systems and Computer Aided Education.
- Miedema, D., Taipalus, T., & Aivaloglou, E. (2023). *Students' perceptions on engaging database domains and structures*. Paper presented at the SIGCSE 2023: Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (pp. 122-128). Association for Computing Machinery.
- Salleh, N. A. B., Lim, A. H.-L., & Chua, F.-F. (2024). *Teaching Elements for Database Courses: An Analysis and Design*. Paper presented at the Proceedings of the 2024 10th International Conference on Frontiers of Educational Technologies.
- Stefanidis, C., & Koloniari, G. (2016). *An interactive tool for teaching and learning database normalization*. Paper presented at the Proceedings of the 20th Pan-Hellenic Conference on Informatics.
- Taipalus, T., & Seppänen, V. (2020). SQL education: A systematic mapping study and future research agenda. *ACM Transactions on Computing Education (TOCE)*, 20(3), 1-33. <https://doi.org/10.1145/3398377>
- Tomas, V., Iserte, S., & Perez, M. (2021). *Learning databases using project-based learning*. Paper presented at the INTED2021 Proceedings, IATED.

- Tomás, V. R., Pérez Francisco, M., & Marqués Andrés, M. (2019). *A Project-based learning for the subject "design and implementation of databases"*. Paper presented at the HEAD'19. 5th International Conference on Higher Education Advances, Editorial Universitat Politècnica de València.
- Valles-Coral, M. A. (2023). Information as a strategic and valuable asset for organizations. *Revista Científica de Sistemas e Informática*, 3(1), e496. <https://doi.org/10.51252/rcsi.v3i1.496>
- Welk, J. (2025). FlippedDB: Applying a flipped classroom approach in a basic database course. *Proceedings of the International Conference on Teaching, Learning and Education*, 2(1), 82–94. <https://doi.org/10.33422/ictle.v2i1.1126>
- Yanti, F. A., & Thohir, M. A. (2024). Higher order thinking skills in science learning: A systematic review from 2014–2023. *International Journal of Evaluation and Research in Education*, 13(4), 2419-2427. <https://doi.org/10.11591/ijere.v13i4.28082>