

Teaching Restructuring and Practical Exploration of Object Engineering Course under the Background of Undergraduate-Graduate Integration

Xiaodong Zhu^{1*}, Shiying Zeng¹, Zihan Shen¹, Jinpeng Sun¹, Xuanyu Shi¹, Yutong Hu¹

¹ Business School, University of Shanghai for Science & Technology, Shanghai, China

Received: 3 November 2025; **Accepted:** 22 December 2025; **Published:** 20 January 2026

Abstract:

Motivated by both the industrialization of the fourth generation artificial intelligence technology and the reform of the undergraduate-graduate integrated education in higher education, Shanghai University of Science and Technology takes the core graduate course “Object Engineering” with a 15 years’ teaching history as the carrier to start the integration reform of undergraduate-graduate in the form of undergraduate AI program, which focus on the programming implementation aspects, yet ignore system design. The important positioning of the curriculum course linking under graduates and enhance graduate students’ abilities and skills of system design is proposed in this paper. Following the overall goals of knowledge integration, ability hierarchy and competency fitting, the reform framework that includes modular curriculum system, differentiated teaching approach and industry-edu-research supporting system is established. The framework can help to face the problems mentioned before regarding the lack of knowledge of the students by using the knowledge gap of bachelor and master studies, the lack of realistic practical situations, that help the students developing their systems thinking and engineering skills. It attempts to nurture artificial intelligence interdisciplinary talents with programming skills, designing abilities, and practical ability, and meanwhile lays down an applied model for integrated undergraduate-graduate building of engineering class curriculum in universities.

Keywords:

Object Engineering, UML, Undergraduate-graduate Integration, Industry-education-research Collaboration, Artificial Intelligence Education

1. Introduction

With the continuous advancement of the new round of technological revolution and industrial transformation, general-purpose technologies such as artificial intelligence are profoundly reshaping the labor market, posing new demands on the scale, structure, and skills of talent [1]. The World Economic Forum points out that AI and digital technologies will significantly change the job structure and increase the demand for highly skilled talent, requiring the education system to adapt to new

trends [2]. Policy is an important guide for AI-enabled talent cultivation. The analysis by the International Labour Organization emphasizes the core role of institutional policies in coordinating the supply and demand relationship between education and the labor market [3]. Against this backdrop, higher education must build a multi-level, high-quality artificial intelligence talent cultivation system, breaking down the barriers between theory and practice, so that graduates possess both a solid academic foundation and engineering practical skills, which is particularly evident in the alignment of university and industry curricula [4]. Exploring strategies for cultivating artificial intelligence talent in universities to meet industrial needs is both a national strategic imperative and a trend of the times. Collaboration among various parties, industry-education integration, and interdisciplinary learning have become important directions for talent cultivation [5].

2. Current Status of Integrated Undergraduate and Graduate Education Practices and the Innovations of this Research

Currently, undergraduate graduates are often limited to the ability to implement single technical modules, while graduate students frequently face a disconnect in transitioning from theory to practical engineering applications. This skills gap between undergraduates and graduates has become a critical bottleneck hindering industrial development and poses a rigid requirement for cross-level integration of university curriculum systems. Therefore, examining the transformation of university talent cultivation structures from the perspective of market supply and demand is not only an inevitable choice for overcoming development bottlenecks in the field of artificial intelligence, but also has universal theoretical and practical value for universities in promoting integrated undergraduate and graduate curriculum reform across all disciplines and adapting to the needs of the new era's industries [6].

Based on this, the author explored curriculum reform in the "Object Engineering" course. As an integrated undergraduate and graduate course, its key lies in bridging the gap for undergraduates from programming skills to systems engineering thinking. Given the reality of continuously expanding undergraduate, master's, and doctoral programs, and the increasing proportion of direct doctoral students, further strengthening the systematic cultivation of students' comprehensive qualities at the undergraduate level is of paramount importance [7]. Some universities, especially many research-oriented universities, now have more graduate students than undergraduates. While meeting the development needs of society and universities to expand enrollment, ensuring the quality of graduate education is particularly crucial. If academic rigor is the lifeblood of university courses, then practicality is their vitality [8]. Thus, students can solidify their professional knowledge during their undergraduate studies and acquire systems engineering thinking during their graduate studies, thereby significantly enhancing their academic research abilities and employability. Universities should continuously adjust and optimize relevant course structures to more effectively address the disconnect between undergraduate and graduate education, genuinely improve the quality of talent cultivation and disciplinary competitiveness, and ultimately promote the organic integration of industry, academia, and research.

Currently, the integrated undergraduate and graduate education in universities exhibits three main characteristics. First, the connection between undergraduate and graduate programs is loose. Some universities only achieve this connection through the simple addition of course content. For example, the existing undergraduate-

graduate integrated model, while emphasizing curriculum continuity, fails to form a structured system of progressive capabilities and lacks a systematic design targeting “basic-advanced-innovative” three-stage capabilities. Second, there is insufficient emphasis on practical application. Relevant curriculum reforms focus on the continuity of theoretical knowledge and fail to deeply integrate into real-world industrial scenarios, leading to a disconnect between students' engineering capabilities and market demands. Third, the collaborative mechanisms are incomplete. Existing training models are mostly limited to resource integration within the university, and industry-university cooperation remains at a superficial level, lacking a deep collaborative mechanism for joint faculty appointments, joint project research, and shared achievements.

Therefore, this paper takes “object engineering” as an example to explore teaching reconstruction in the context of integrated undergraduate and postgraduate education, mainly focusing on the following aspects. First, it proposes a disciplinary framework of “three-dimensional objectives, four major principles, and three-level modules,” abandoning the shortcomings of the previous loosely connected and disjointed disciplines. It clarifies a hierarchical training program where undergraduate education serves as the foundation and connection, postgraduate education focuses on advancement and deepening, and interdisciplinary studies emphasize innovation and expansion, thus achieving precise positioning and coupling of knowledge continuity and sequential ability development. Second, it proposes an interdisciplinary integrated education model of “technology, management, and engineering,” coordinating the faculty resources of computer science and management science and engineering. It systematically designs a curriculum that combines management thinking and process optimization, building a new collaborative education model while meeting the multifaceted needs of complex software system development. Third, it constructs an integrated training mechanism of “university-enterprise dual mentors, real projects, and progressive practical training,” promoting substantive enterprise participation in all aspects of teaching and research, including curriculum design, practice, and assessment. This effectively addresses the problem of the disconnect between theory and industry needs in traditional models. Fourth, it adopts differentiated and hierarchical teaching measures. This differentiation is reflected in theoretical teaching, case analysis, experiments and internships, and comprehensive projects, addressing the problem of rigid theoretical and practical teaching caused by the previous “one-size-fits-all” teaching model.

3. Analysis of the Characteristics of the Integrated Undergraduate and Graduate Course in Object Engineering

3.1. Core Positioning of the Course

Connecting the undergraduate course “Advanced Computer Programming” and the cultivation of graduate students' system design capabilities, “Object Engineering” plays a crucial role in bridging the gap, accumulating skills, inheriting content, and transforming thinking. The rapid advent of the artificial intelligence era has made computer programming an important general course. It is one of the basic abilities of technical application talents and an essential path for students to develop innovative thinking and solve complex problems[9]. The undergraduate course “Advanced Computer Programming” has already equipped students with basic knowledge and skills in applying advanced computer programming languages and implementing simple algorithms. “Object Engineering” connects the course content and knowledge

points of the undergraduate program to the complex requirements of graduate studies, such as the implementation of large-scale systems for artificial intelligence, system framework design, module construction, system performance tuning, and optimization. Under the three-tiered design of “strengthening foundations - enhancing capabilities - fostering innovation,” it facilitates the transformation from a simple programmer to a developer of complex systems.

3.2. Three-Dimensional Teaching Objectives

(1) **Disciplinary Integration Objectives.** Interdisciplinary integration is one of the key focuses of current educational reform. Educators must grasp the essence of interdisciplinary integration and find its logical starting point [10]. The “Object Engineering” course will be deeply integrated with the artificial intelligence professional curriculum system, serving as an extension and advanced course to “Advanced Computer Programming,” cultivating students' abilities in object-oriented analysis, design, and implementation in the development of artificial intelligence application systems.

(2) **Theoretical Methods Objectives.** “Object Engineering” focuses on object-oriented system analysis and design theory as its core teaching content, emphasizing the cultivation of UML modeling and expression abilities. Through a combination of theoretical explanations and method demonstrations, it helps students accurately master abstract modeling methods and build a mature system thinking framework when dealing with artificial intelligence system development problems.

(3) **Skills Practice Objectives.** “Object Engineering” uses case studies as a pathway and project practice as a vehicle, focusing on the step-by-step improvement of students' engineering capabilities. It encourages students to use divergent thinking to select intelligent scenarios from various industries as research subjects, gradually improving their system capabilities. Through these application scenarios, it strengthens students' core abilities in requirements analysis, architectural design, and teamwork, cultivating outstanding talents with independent research capabilities and innovative spirit.

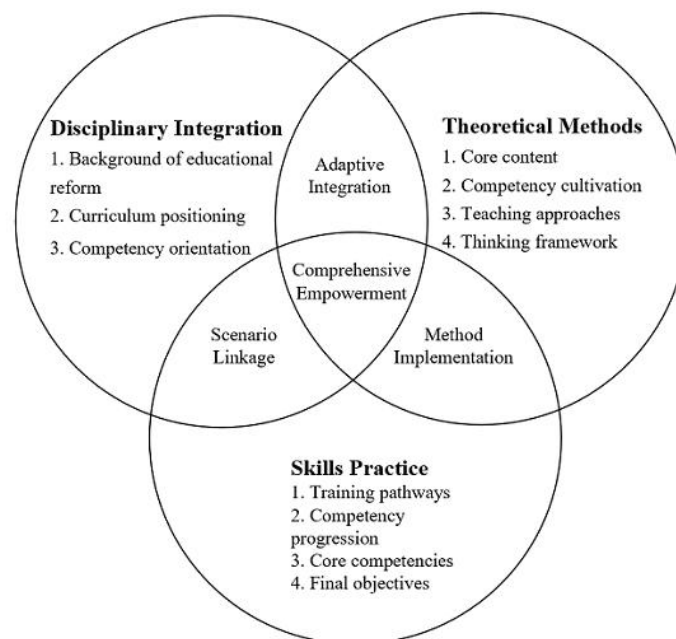


Figure 1. Three-Dimensional Instructional Objectives.

3.3. Four Guiding Principles of Curriculum Development

(1) Systemic Principle. Based on the knowledge system and capability development logic of “Object-Oriented Engineering, a complete curriculum system with gradually increasing difficulty is constructed. This system integrates teaching elements horizontally, including theoretical lectures, case studies, project practice, and outcome assessment. The undergraduate stage focuses on programming skills development, while the graduate stage focuses on the architectural optimization and innovative design of complex object systems, forming a complete chain of engineering capability development.

(2) Hierarchical Principle. Teaching content and tasks are designed in layers based on the cognitive levels and core capabilities of undergraduate and graduate students. The foundational stage strengthens the connection between undergraduate and graduate knowledge points, such as basic concepts and core modeling methods, and reinforces them through simple object module development tasks. The advanced stage focuses on the application of design patterns and architectural decomposition methods for complex object systems, requiring students to complete medium-scale object-oriented engineering projects. The innovation stage encourages graduate students to conduct personalized exploration in conjunction with research topics or industry needs, designing innovative solutions to achieve customized teaching.

(3) Practicality Principle. The practicality principle is based on real industry needs, emphasizing the fundamental role of practical teaching. By introducing real enterprise project cases, a school-enterprise cooperation platform is formed, allowing students to participate in the requirements design, module development, and testing phases of actual object-oriented engineering projects, creating a seamless process from course design to graduation projects and research projects. Undergraduate course design revolves around the development of basic object modules, while graduate graduation projects integrate complex object-oriented system innovation projects, ensuring that educational outcomes align with the industry's skill requirements for object-oriented engineering talents.

(4) Collaborative Principle. A multi-dimensional collaborative development mechanism is established. First, teachers collaborate to integrate the advantages of the undergraduate teaching team and the graduate supervision team, forming a “basic teaching + advanced guidance” dual-mentor model to jointly guide student practical projects. Second, school-enterprise collaboration involves jointly developing course content with technology companies, establishing practical bases, and inviting enterprise engineers to participate in classroom teaching, ensuring that teaching updates are synchronized with industry technological developments. Third, teacher-student collaboration encourages students to participate in case selection and practical task design. Teaching content and methods are continuously optimized based on student feedback, thus forming a virtuous cycle of mutual learning.

3.4. Key Problems to be Addressed

(1) Disconnect between undergraduate and graduate student skill development. Undergraduate courses primarily focus on programming skills, while graduate courses emphasize systematic design and innovative research capabilities. The lack of continuity in curriculum design prevents students from smoothly transitioning from “programming implementation” to “system design” capabilities, resulting in repetitive and disjointed course content between undergraduate and graduate levels.

(2) Shortcomings in training due to an overemphasis on programming and neglect of design. Although artificial intelligence students are proficient in programming languages such as Python and C++, they lack systematic training in object-oriented modeling concepts and engineering analysis and design methodologies. This hinders their cognitive leap from “writing code” to “architectural design” in complex software system development.

(3) Disconnect from industry needs. The industrialization of next-generation artificial intelligence urgently requires versatile talents who are proficient in both programming and complex system engineering design. Existing practical courses remain disconnected from real-world enterprise development environments, leading to a gap between students’ engineering thinking, teamwork abilities, and industry demands. Therefore, there is an urgent need to enhance their employability.

4. Reform Pathways for Object-Oriented Engineering Courses Integrating Undergraduate and Graduate Education

4.1. Modular Curriculum System Reconstruction

Following the principles of “foundation transition, gradual deepening, and spiral progression,” this framework breaks down the barriers between undergraduate and graduate courses and establishes two basic modules.

(1) Undergraduate Foundation Connection Module. Based on the “Advanced Computer Programming” course, this module focuses on “knowledge foundation and skill enlightenment.” The core content includes basic theories, core syntax of modeling languages and operation of common tools, and analysis and design of simple software systems. Through modular design, it ensures that students master the basic knowledge points for transitioning from programming skills to system design capabilities, building a bridge for learning in the graduate stage.

(2) Graduate Advanced Enhancement Module. Focusing on “complex system design and innovative applications,” this module aims for a comprehensive improvement of knowledge and abilities. The teaching content mainly includes advanced UML modeling techniques, architecture and detailed design of complex software systems, system engineering methods in intelligent environments, and cross-disciplinary project collaborative design. This module is designed to address industry needs, comprehensively improving students’ R&D and design capabilities in areas such as recommendation systems, intelligent healthcare, and other complex artificial intelligence application case studies, as well as designing and optimizing complex systems.

(3) Interdisciplinary Spiral Upgrade Module. By utilizing the same course units to avoid simple repetition of course content, a spiraling upward relationship of the course unit system is formed—“undergraduate degree courses lay the foundation, and master's degree courses broaden the scope”—thus ensuring the systematic continuity and consistency of learning in both undergraduate and master’s degree courses.

4.2. Differentiated Teaching Strategies Implementation

To meet the different cognitive levels and ability requirements of undergraduate and graduate students, differentiated teaching strategies were implemented, covering four core teaching aspects.

(1) Layered theoretical instruction. Undergraduate teaching focuses on general concepts, providing simple explanations of basic concepts, standards, or principles, accompanied by simple case studies; graduate teaching focuses on the principles of complex system design, primarily using artificial intelligence technology and object-oriented engineering methods. Undergraduate students with strong academic abilities are also encouraged to take the “Advanced Computer Programming” course and then select the graduate course “Object-Oriented Engineering,” with credit transfer options available.

(2) Layered case analysis. Undergraduate teaching uses small software systems, such as library management software, for basic requirements analysis and model creation; graduate teaching uses commercial cases, such as the design of intelligent recommendation applications and intelligent healthcare data platforms, to cultivate critical thinking skills through in-depth analysis and discussion of these cases.

(3) Layered experimental practice. Undergraduate students conduct practical experiments in computer labs to ensure mastery of basic skills; graduate students conduct comprehensive experimental projects in laboratories and utilize virtual simulation platforms to simulate enterprise development processes.

(4) Layered team projects. Undergraduate students complete small projects in groups of 3-4 people, focusing on basic collaboration and process familiarity; graduate students undertake real-world enterprise projects or research topics in teams of 5-6 people to strengthen engineering capabilities and innovative thinking.

4.3. Support System Construction

4.3.1. Faculty Team Building

(1) Teacher Professional Skills. Regularly offer professional training in object-oriented system development, advanced UML applications, and artificial intelligence system design. Invite industry professionals to participate in professional teaching research to improve teachers' professional level. Conduct teaching experience and case sharing activities through sharing courseware and unified collective lesson preparation.

(2) Interdisciplinary Collaboration Mechanism. Establish an interdisciplinary teaching team composed of computer science, management science, and engineering teachers, clearly defining collaborative functions and work processes. Computer science teachers are responsible for technical theory and practical teaching, while management science and engineering teachers focus on management thinking and process optimization in system design, forming a complementary teaching system.

4.3.2. Industry-Education-Research Collaborative System

(1) Deep Integration of School and Enterprise. First, select affiliated enterprises with strong technical research and development capabilities and abundant practical resources. Organically integrate real-world work project cases and cutting-edge technical information into classroom teaching. Invite experts with practical R&D project experience from enterprises to participate in relevant teaching, ensuring that the content of course teaching aligns with industry needs.

(2) Practical Training Base Construction. Jointly build on-campus and off-campus practical training bases to provide students with a realistic system development

environment. Undergraduate students learn about industry processes through company visits and short-term internships; graduate students participate in core project development within enterprises, following the entire process of requirements analysis, model selection, and system deployment to accumulate practical experience.

(3) Enterprise Mentor Participation. Recruit relevant key personnel from enterprises as part-time mentors, providing guidance to students on project operation and career development through online and offline instruction. In the process of project evaluation, a joint evaluation by on-campus teachers and enterprise mentors is adopted to ensure that the quality of student training meets industry technical standards.

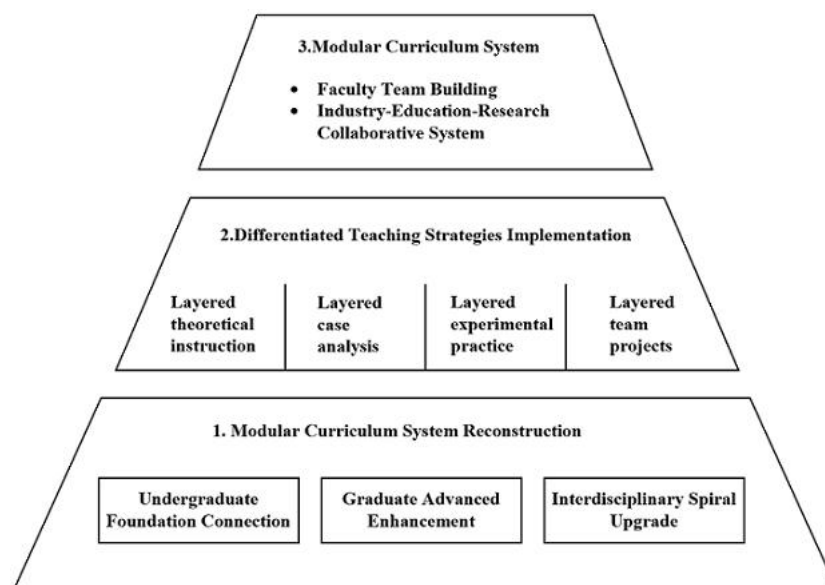


Figure 2. Reform implementation paths.

5. Expected Outcomes of the Reform

Through three years of systematic implementation, this reform will achieve significant breakthroughs in knowledge integration, skill enhancement, and curriculum evaluation, creating a model for integrated undergraduate and graduate talent cultivation.

In terms of knowledge connection, the reform will break down content and ability barriers by addressing the differences in course content and abilities between undergraduate and graduate programs. It will build a main thread of concept, ability, and embedded skills, enabling students to progress from “being able to program” to “being able to implement” and then to “being able to implement systematically.” This will ensure a coherent knowledge system and a tiered approach to ability development, reducing learning gaps during the transition from undergraduate to graduate studies.

Regarding ability enhancement, students will receive systematic training in object-oriented programming, analytical modeling, analytical design, and design implementation during their studies. They will acquire practical work experience in module development, teamwork, and project organization in a real-world industrial environment. The curriculum will utilize layered practice, project-based learning, and industry internships to enhance students’ engineering thinking, innovation and entrepreneurship abilities, and professional skills. Students will be encouraged to

participate in various academic competitions, research training, and entrepreneurial projects, creating an artificial intelligence talent cultivation model driven by projects and abilities.

In terms of teaching assessment and evaluation, a multi-party evaluation system involving students, colleagues, enterprises, and experts will be established, with comprehensive coverage, continuous monitoring, and feedback loops. This system will shift from process monitoring to outcome and output monitoring. Based on the achievement of course objectives, assessment indicators will be established for knowledge mastery, ability enhancement, thinking development, and career preparation. Timely quantitative sampling and comparative analysis will be conducted, presented in charts, and fed back to students and teachers as a reference for continuous course improvement.

Once the reformed curriculum matures, it is expected to be nominated as a top-tier course at the university and provincial levels, and as a candidate for teaching achievement awards. It will serve as a replicable and scalable model for integrated undergraduate and graduate curriculum development, providing practical exploration for the construction of artificial intelligence-related professional curriculum systems and promoting higher engineering education to a higher level.

6. Conclusions

This paper focuses on the systematic restructuring and practical implementation of object-oriented engineering courses within the context of integrated undergraduate and graduate education. Addressing prominent issues such as talent gaps, fragmented teaching, and disconnects with industry needs in the process of cultivating AI talent, this paper explores an integrated reform design based on three-dimensional objectives, four principles, a modular system, differentiated teaching, and industry-academia collaboration. After three years of phased planning, this course has largely completed its restructuring in terms of teaching content, teaching methods, practical systems, faculty, and assessment, initially achieving a continuous training path—a seamless connection from undergraduate “programming” to graduate-level “system design”—bridging the “gap” in talent cultivation and providing students with a systematic, open, and industry-relevant learning experience.

In the future, we will continue to advance course reform in three directions. intelligence, systematization, and outreach, constantly exploring and experimenting to make object-oriented engineering courses more standardized, widely applicable, and effective. Integrating artificial intelligence into object-oriented engineering course teaching will enable intelligent teaching of object-oriented engineering; course development and research will transition from case studies to industry applications, forming a replicable, adaptable, and scalable course system standard and teaching guide; ultimately, continuously tracking and meeting technological advancements, industrial transformation, and market demands, constantly improving course content and teaching methods, and striving to create a model for undergraduate and graduate talent cultivation in artificial intelligence courses, making due contributions to building a high-level higher education system and achieving China’s goal of becoming an educational powerhouse.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Funding

This article is supported by the 2025 Graduate Teaching Project of University of Shanghai for Science and Technology (Undergraduate-Graduate Integrated Curriculum on “Object Engineering”).

References

- [1] Lassébie, J. Skill needs and policies in the age of artificial intelligence. OECD Employment Outlook, 2023; pp. 155.
- [2] Di Battista, A.; Grayling, S.; Hasselaar, E. et al. Future of jobs report 2023[C]//World Economic Forum. 2023; pp. 978-2.
- [3] Sapawi, M.S.M.; Yusoff, N.M.R.N. Artificial Intelligence in Curriculum Development. A Global Systematic Review of Trends, Challenges, and Strategic Directions. *Journal of Curriculum Studies Research*, 2025, 7(2), 466-497.
- [4] Wu, J.C.K.; Lin, T.W.; Wang, F.G. AI Talent Cultivation. Higher Education-Industry Alignment in Taiwan. 2025.
- [5] Castonguay, Y.; Hamouti, R. Artificial Intelligence. A Lever for University-Industry Collaboration—An Exploratory Review. *Int. J. Learn. Teach*, 2025, 11. 212-215.
- [6] Shabangu, N.; Madondo, N.E. Addressing graduate unemployment. Aligning higher education curricula with labour market demands[C]//The Focus Conference (TFC 2024). Atlantis Press, 2024; pp. 74-87.
- [7] Huerta-Gomez-Merodio, M.; Requena-Garcia-Cruz, M.V. Integrating theory and practice in engineering education. A cross-curricular and problem-based methodology. *Education Sciences*, 2025, 15(9). 1253.
- [8] Matthews, K.E.; Tai, J.; Enright, E, et al. Transgressing the boundaries of ‘students as partners’ and ‘feedback’discourse communities to advance democratic education. *Teaching in higher education*, 2023, 28(7), 1503-1517.
- [9] Carney, S. Reimagining our futures together. a new social contract for education. by UNESCO, Paris, UNESCO, 2021; pp. 186 pages, ISBN 978-92-3-100478-0. 2022.
- [10] Huerta-Gomez-Merodio, M.; Requena-Garcia-Cruz, M.V. Integrating theory and practice in engineering education. A cross-curricular and problem-based methodology. *Education Sciences*, 2025, 15(9), 1253.



© 2026 by the author(s); licensee International Technology and Science Publications (ITS), this work for open access publication is under the Creative Commons Attribution International License (CC BY 4.0). (<http://creativecommons.org/licenses/by/4.0/>)