Journal of Software Engineering Research and Development, 2026, 14:29, doi: 10.5753/jserd.2026.7768
This work is licensed under a Creative Commons Attribution 4.0 International License.

An analysis of Brazilian Software Engineering undergrad courses in public and private education institutions

Izadora Lima [**Universidade Federal do Maranhão** | ***izadora.paula@discente.ufma.br***]
Marcos Oliveira [**Universidade Federal do Maranhão** | ***marcos.jansem@discente.ufma.br***]
Bruno Silva [**Universidade Federal do Maranhão** | ***bruno.carvalho1@discente.ufma.br***]
Fernando Ayach [**Universidade Federal do Mato Grosso do Sul** | ***fernando.ayach@ufms.br***]
Diogo Menezes [**Universidade Federal do Mato Grosso do Sul** | ***diogo.menezes@ufms.br***]
Rodrigo Santos [**Universidade Federal do Estado do Rio de Janeiro** | ***rps@uniriotec.br***]
Awdren Fontão [**Universidade Federal do Mato Grosso do Sul** | ***awdren.fontao@ufms.br***]
Davi Viana [**Universidade Federal do Maranhão** | ***davi.viana@ufma.br***]

Abstract

The teaching of Software Engineering (SE) in Brazil began in subjects within Computer Science courses. SE education evolved with the creation of specific courses, due to the launch of the first guidelines from the Association for Computing Machinery (ACM), references from the Brazilian Computer Society (SBC), and the National Curriculum Guidelines (DCNs). Despite this progress, systematic analyses of the academic profiles of these courses remain scarce. This study investigates the curriculum structure of 28 undergraduate courses in SE offered by public higher education institutions (HEIs) and 35 undergraduate courses in SE offered by private institutions in Brazil, based on the SBC's training axes and the areas of the Software Engineering Body of Knowledge (SWEBOK), through an analysis of the syllabi of each course. Regarding public institutions, the data reveal a wide geographical distribution, with a predominance in the non-capital cities. Subjects related to the Fundamentals axes appear frequently, whereas critical areas such as Quality, Requirements, and Software Testing receive comparatively limited formal coverage. Regarding private institutions, the data show an uneven geographical distribution, with a strong concentration of courses in certain regions of the country. Differences in distribution between capitals and the non-capitals are also observed. In terms of curriculum, there is a predominance of subjects related to computing fundamentals, software construction, and professional practices. In contrast, areas such as quality, maintenance, and software architecture are less covered. Inferential analyses showed that public and private programs have significantly different geographic distributions, with private programs concentrated mainly in the South and Southeast. However, no statistically significant regional variation was found in SWEBOK areas coverage within either sector. Actions such as revising the course programs, encouraging applied research in Software Engineering, and making the curriculum more flexible with different emphases can mitigate the asymmetries in Software Engineering education across the country. This article extends a previous work recognized with the Best Paper Award at SBES 2025 Education Track, which analyzed only public institutions. By incorporating private institutions into the analysis, this extended version provides a broader and more comprehensive picture of Software Engineering education in Brazil.

Keywords: *Teaching Software Engineering, Analysis of syllabi, SWEBOK*

1 Introduction

The quality of software systems, which are increasingly present in critical contexts of everyday life, depends directly on the solid and up-to-date training of the professionals involved in their development (Shaw, 2000). A noteworthy aspect is that, to ensure student success, software engineering (SE) curricula should be of high quality, aligned with industry requirements, and provide the skills and knowledge most valued by industry (Mishra and Mishra, 2021). In this scenario, SE education has been gaining ground as an autonomous field of knowledge with its own educational challenges (Ferreira et al., 2018).

Historically, Brazilian courses have trained these professionals primarily through traditional programs in Computer Science, Computer Engineering, and Information Systems. However, the growing perception that existing courses did not adequately prepare students for the software industry led to the emergence of specialized software engineering courses. Initially as isolated subjects and later as independent undergraduate programs (Wangenheim and Silva, 2009; da C. Figueiredo et al., 2010). The need for training better aligned with the software industry drove this evolution. In this sense, the Brazilian Computer Society (SBC) played a fundamental role in proposing benchmarks for SE training, aiming to guide the structuring of undergraduate courses in the area (SBC, 2017).

Software Engineering education has evolved significantly in response to the growing complexity of software systems and the increasing demand for qualified professionals. Universities have been challenged to design curricula that not only cover theoretical foundations but also foster practical skills such as collaboration, problem-solving, and the ability to address real-world constraints. In this context, several international initiatives, such as the curriculum guidelines proposed by the Association for Computing Machinery (ACM) and the IEEE Computer Society (Ardis et al., 2015), have sought to establish reference frameworks for structuring SE programs and courses. These guidelines emphasize the need to balance fundamental concepts, engineering practices, and

professional competencies, thereby reinforcing the role of higher education institutions in preparing students to address the technological and societal challenges of modern software development.

Despite these advances, there is still recurring criticism regarding the depth with which fundamental SE topics are addressed in curricula, especially about the integration of theory and practice, the adequate treatment of areas such as "Project Management", "Software Testing", "Software Requirements", and "Software Architecture", and adherence to national and international standards (Wangenheim and Silva, 2009; Ferreira et al., 2018). In addition, new software industry demands, such as agile methodologies, software engineering for startups, sustainability, ethics, and artificial intelligence, reveal gaps that curriculum structures have not yet fully addressed (Ferreira et al., 2018; Betz and Penzenstadler, 2025).

Taking this into account, this article investigates how Software Engineering undergraduate programs are organized and distributed among public and private higher education institutions in Brazil. It provides a curriculum-level, nationwide analysis of 28 public and 35 private programs, examining their alignment with both the SBC reference curriculum and the SWEBOK V4.0 knowledge areas, and mapping the presence of main topics and potential curricular gaps. By offering a dual-framework and sector-sensitive view of the current curricular landscape, this study advances the literature on Software Engineering education in Brazil and provides data-based overview to support the continuous improvement and national standardization of SE curricula.

Initially, Oliveira et al. (2025) analyzed 28 Software Engineering programs offered by public institutions in Brazil and found a balanced regional distribution, with a higher concentration in the North and a lower presence in the Southeast. Most programs are located in non-capital cities rather than state capitals, except in the Midwest region, where a greater number of programs are concentrated in capital cities. An increase in the number of programs offered between 2008 and 2024 was also observed. Regarding the curricular structure and its alignment with SWEBOK, courses related to the Fundamentals axis (42.2%) and Software Construction and Testing (19.8%) were predominant, while Software Quality received limited coverage (3.5%). The regional analysis showed that areas such as Professional Practice and Software Construction are more frequently addressed, whereas Models and Methods, Software Maintenance, and Software Security have lower coverage. Compared with international curricula, Brazilian programs place a stronger emphasis on Computing Fundamentals and Software Engineering Economics, but show a weaker presence of advanced topics and software quality. These results indicate progress in Software Engineering education in Brazil; however, they also highlight gaps in areas essential to professional practice.

While the analysis of public institutions offers a view of federally and state-funded SE education, Brazil's higher education landscape is predominantly private, with private institutions accounting for the majority of enrolled students nationwide. Excluding private institutions from the analysis thus leaves a significant portion of SE education unexamined. This extended version addresses that gap, enabling, for the first time, a comparative analysis of public and private SE curricula in Brazil.

In this extended version, we included and analyzed 35 Software Engineering course programs offered by private institutions in Brazil. We observed an uneven regional distribution, with a strong concentration of courses in the South and Southeast. We also identified that, in the South and Southeast regions, most courses are located in inland cities, while in the North and Northeast regions, courses are predominantly concentrated in capital cities. In terms of curriculum structure, there was a predominance of subjects associated with the fundamentals of computing and software development, accompanied by content related to professional practice, while areas such as software quality and maintenance were less represented in the curricula analyzed.

We also compared public and private course programs and identified significant curricular imbalances. In both public and private institutions, a large proportion of courses focus on computing fundamentals and practical implementation, while key Software Engineering areas such as Software Quality, Requirements Engineering, Software Testing, and Software Architecture are less frequently addressed. Additionally, the recurrent underrepresentation of several core SE areas across the analyzed programs indicates uneven formal curricular coverage at the national level. These findings highlight the need for curriculum revisions and stronger alignment with international guidelines and industry needs in order to improve the training of future software professionals.

The remainder of this article is organized as follows. Section 2 presents the theoretical background and related work, including the SWEBOK knowledge areas, the SBC reference curriculum, and international curriculum models for Software Engineering education. Section 3 describes the research method adopted in this study, including the data collection process and the analysis procedures. Section 4 presents the results of the analysis of Software Engineering undergraduate programs in Brazil. Section 5 discusses the implications of the findings and their impact on Software Engineering education. Section 6 discusses threats to validity. Finally, Section 7 concludes the article and outlines directions for future work.

2 Background and Related Work

This section presents the theoretical foundations that underpin the analysis of the curriculum organization of Software Engineering undergraduate programs in Brazil. The discussion draws on the reference curriculum established by the Brazilian Computer Society (SBC), the SWEBOK, and related studies on professional training and competency development in the field.

2.1 SWEBOK

The Guide to the Software Engineering Body of Knowledge (SWEBOK Guide V4.0), published by the IEEE Computer Society in 2024, consolidates accepted knowledge in software engineering and serves as a reference for education, professional certification, and software engineering practice.

The SWEBOK organizes knowledge into 18 Knowledge Areas (KAs), which represent the main theoretical and practical aspects of software engineering (IEEE Computer Society, 2024). The areas are as follows:

- Area 1 - Software Requirements: addresses the elicitation, analysis, specification, validation, and management of software requirements. Correctly identifying and documenting requirements is fundamental to the success of projects;
- Area 2 - Software Architecture: refers to the definition of the organizational structure of software systems, covering the breakdown into components, the relationships between them, and the principles and guidelines that guide their design and evolution;
- Area 3 - Software Design: discusses the transformation of requirements into viable solution models, including architectural decisions, component design, and detailing of algorithms and data structures;
- Area 4 - Software Construction: focuses on software implementation, including programming practices, component integration, and basic verification. It highlights the importance of code quality and adherence to good development practices;
- Area 5 - Software Testing: encompasses the principles, techniques, and methods for software verification and validation, including unit, integration, system, and acceptance testing, to ensure product quality and compliance with requirements;
- Area 6 - Software Engineering Operations: addresses activities related to the operation and support of systems in production environments, including deployment, monitoring, corrective and preventive maintenance;
- Area 7 - Software Maintenance: deals with the modification of software systems after their initial delivery, whether to correct defects, improve performance, or adapt the system to new requirements or environments;
- Area 8 - Software Configuration Management: refers to the control and management of changes to software artifacts, including identification, version control, auditing, and release of configurations;
- Area 9 - Software Engineering Management: covers activities related to planning, organizing, coordinating, and monitoring software projects, aiming to ensure that quality, deadline, and cost objectives;
- Area 10 - Software Engineering Process: deals with the definition, implementation, and continuous improvement of software development processes, involving traditional life cycles and agile methods;
- Area 11 - Software Engineering Models and Methods: explores the use of formal and informal approaches to software development, such as prototyping, iterative, heuristic, and rigorous process-based methods;
- Area 12 - Software Quality: discusses quality assurance and control practices in software products, including metrics, quality assessments, audits, and continuous improvement;
- Area 13 - Software Security: addresses principles and practices to ensure that software is resistant to threats, vulnerabilities, and attacks, from the requirements phase through design, implementation, and security testing;
- Area 14 - Professional Practice of Software Engineering: focuses on ethical, legal, social, and professional aspects related to the practice of Software Engineering, including interpersonal skills, leadership, and social responsibility;
- Area 15 - Software Engineering Economics: applies economic principles to Software Engineering, addressing cost estimation, feasibility analysis, and decision support in projects;
- Area 16 - Computing Foundations: covers essential fundamentals of Computer Science relevant to the practice of Software Engineering, such as algorithms, data structures, databases, networks, and distributed systems;
- Area 17 - Mathematical Foundations: deals with the mathematical foundations that underpin Software Engineering, including mathematical logic, algebra, statistics, and graph theory;
- Area 18 - Engineering Foundations: presents fundamental engineering concepts applied to software practice, such as modeling, analysis, experimentation, and empirical methods for evaluating solutions.

The SWEBOK Guide V4.0 (IEEE Computer Society, 2024) consolidates fundamental knowledge of software engineering, updating its areas to incorporate modern practices such as agile development, DevOps, and emerging technologies, and serves as a reference for training professionals in the field.

2.2 SBC Reference Curriculum for Software Engineering

The Brazilian Computer Society (SBC) has proposed a reference curriculum to guide the structuring of Software Engineering courses in Brazil, aiming to consolidate pedagogical practices aligned with the professional and academic demands of the field (SBC, 2017). This curriculum was developed by experts and professors in the field, based on the National Curriculum Guidelines (NCGs) and international recommendations from the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE), especially the Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering (Ardis et al., 2015).

The proposal organizes software engineering education into seven main axes. Each axis is associated with competencies, skills, and content that courses should develop in an integrated manner throughout the program. Below, we describe the main areas defined in the SBC reference curriculum:

- Axis 1 - Fundamentals of Computing, Mathematics, and Production: comprises the theoretical and technical knowledge essential for solving computational problems, including algorithms, logic, discrete mathematics, data structures, algorithm complexity, computer architecture, and fundamentals of systems engineering;
- Axis 2 - Entrepreneurship and Innovation: addresses the development of skills to identify opportunities, plan innovative businesses, raise funds, and manage ventures,

encouraging market-oriented action and the generation of original solutions;
- Axis 3 - Complementary Professional Skills and Practices: covers fundamental aspects for the student's comprehensive education, such as effective communication, teamwork, negotiation, ethics, intellectual property, consulting, and scientific research, preparing them for diverse professional contexts;
- Axis 4 - Software Management and Process: focuses on modeling, applying, and continuously improving software processes. It includes practices related to project management, software life cycles, evolution, operation, metrics, and tools to support software engineering.
- Axis 5 - Software Requirements, Analysis, and Design: deals with the elicitation, modeling, and validation of requirements, as well as the architectural and detailed design of systems, with an emphasis on understanding the contexts of use and the needs of users and stakeholders;
- Axis 6 - Software Construction and Testing: covers the implementation, integration, and verification of software systems, using programming techniques, static and dynamic testing, frameworks, reuse, and secure development practices;
- Axis 7 - Software Quality: emphasizes the production of reliable, high-quality software through metrics, standards, quality models, and preventive strategies to detect failures, especially in critical systems.

In addition to being organized by themes, the curriculum proposes a learning taxonomy that distinguishes between three levels of mastery (knowing, understanding, and applying). This taxonomy aims to support instructional planning by enabling the definition of more precise, more measurable learning objectives that align with the progressive development of expected competencies.

2.3 International Reference Curricula

According to Dierl et al. (2021), SE course curricula have been designed with three philosophies in mind. The first treats SE as an additional area to Computer Science programs. In this type of curriculum, the recommendations of the German Computer Society Dierl et al. (2021) are particularly notable.

The second philosophy considers software engineering as engineering for computers, in which the curriculum is derived from other engineering disciplines, and a large part of the course is devoted to multidisciplinary science and computational mathematics. Programs in India All India Council for Technical Education (2018) and Russia Dierl et al. (2021) are notable examples of this type of philosophy.

The last philosophy indicates that software engineering requires prior knowledge of the domain. This philosophy means that, for proper learning, students must specialize in an application domain. For this type, the ACM/IEEE Joint Task Force on Computing Curricula Dierl et al. (2021) curriculum stands out.

All of these international curricula organize their courses into the following areas of knowledge: mathematical foundations; fundamentals of computer science and programming; core software engineering; advanced topics and fundamentals of engineering; economics, ethics, and management; and, finally, technical writing and presentation. We compare the definitions of these areas with the SWEBOK areas and the SBC axes, and show the equivalence in Table 1.

2.4 Related Work

The literature presents a study that reviews undergraduate computer science curricula to identify gaps between teaching and practice. In Azevedo and da Silva (2023), the authors analyzed the curricula of data engineering courses. The analysis of 108 subjects revealed an excessive focus on traditional fundamentals, such as relational databases and ER modeling, and a lack of up-to-date, industry-valued content, such as big data, cloud computing, and data pipelines. Although this is another field, the results reinforce the urgency of curriculum reviews in specific areas of Computer Science, such as Software Engineering.

The literature highlights several challenges in teaching Software Engineering, particularly in aligning training with industry demands. As early as 2000, Shaw (2000) noted that traditional computer science education did not adequately address the fundamental principles and practices of SE and advocated reformulating curricula to emphasize practical skills, modeling, software processes, and requirements engineering.

Several studies have analyzed undergraduate software engineering curricula. In Radaideh (2021a,b), for example, the authors evaluated the software engineering curriculum at a Jordanian university based on the knowledge areas of SWEBOK-V3.0. The authors used a four-point scale to measure the adherence of each area, ranging from "fully compliant" to "minimally compliant." The results indicated full compliance with requirements, design, and software testing, while construction and maintenance were only partially compliant.

Zarour et al. (2025) propose a framework for analyzing higher education course curricula, which is applied in a case study involving nine universities in Saudi Arabia. The framework consists of five steps: (i) understanding SWEBOK V4.0 by evaluators; (ii) defining the scope of evaluation, which may include curriculum, teaching methods, faculty, infrastructure, and student outcomes; (iii) defining the evaluation instrument based on the scope; (iv) conducting and validating the assessment, with specific approaches according to the scope; and (v) presenting the results and recommendations. The application of the framework revealed a predominance of content focused on computing fundamentals, mathematics, and professional practices, and a reduced presence of topics related to software processes and quality.

Studies also highlight gaps between academic training and the skills required by the market. Wangenheim and Silva (2009), in their study of professionals in the software engineering sector, identified the importance of topics such as Project Management, Software Testing, and Software Requirements. However, the authors highlight gaps between what is taught in undergraduate programs and what is actually required in professional practice.

Table 1. Relationship between knowledge areas in international curricula, SWEBOK areas, and SBC axes.

Knowledge Area	**SWEBOK Area**	**SBC Axis**
Mathematical Foundations	17	1
Computer Science and Programming Foundations	4, 6, and 16	1 and 6
Software Engineering Core	1, 2, 3, 4, 6, 7, 8, 10, 11, 12, and 13	4, 5, and 7
Advanced Topics and Engineering Foundations	14 and 18	1
Economics, Ethics, and Management	9, 14, and 15	2, 3, and 4
Technical Writing and Presentation	14	3

Garousi et al. (2020) conducted a systematic literature review focusing on studies that report gaps between the content taught in software engineering courses and industry demands. To identify these gaps, the areas defined by SWEBOK V3.0 were considered. The authors also employed a quadrant-based approach to assess the industry's relative importance of specific knowledge against the gap observed in academic training. In the studies analyzed, covering the period from 2005 to 2018, the areas of "Requirements", "Configuration Management", "Testing", "Quality", and "Design" were classified as highly important for the industrial sector and presented high knowledge gaps. On the other hand, the area of "Mathematical Fundamentals" was considered of low importance and presented a reduced gap.

In the Brazilian context, da C. Figueiredo et al. (2010) investigated the institutionalization of undergraduate courses in SE, analyzing the creation of the first specific undergraduate courses beginning in 2008 and discussions with the Ministry of Education on standardizing course nomenclature. The authors defend the autonomy of SE as a distinct area from Computer Engineering, with its own guidelines and curricular structures.

Similarly, Ferreira et al. (2018) observed the absence or superficial treatment of emerging topics, such as agile methods and software architecture, in the compulsory subjects of many courses, as well as difficulties in teaching practices, infrastructure, and teacher training.

Clementino et al. (2024) conducted a systematic mapping of publications on SE teaching in the SBES proceedings between 2013 and 2023. The study identified project-based teaching approaches, gamification, and active methodologies as predominant, with a focus on practical activities in subareas such as software analysis and design, development processes, and testing. However, the research did not directly address the formal structure of the curricula or their adherence to benchmarks such as the SBC curriculum or SWEBOK, a gap this work seeks to fill.

Unlike previous studies, which explore isolated topics or propose specific pedagogical approaches, our analysis offers a nationwide, comparative perspective on curricular organization across institutional sectors, thereby expanding the empirical basis for discussions of SE education in Brazil.

To the best of our knowledge, no prior study has systematically compared Software Engineering undergraduate curricula across public and private institutions in Brazil using both SBC and SWEBOK as complementary analytical lenses. This study advances the literature by filling that gap with a dual-framework and geographically informed analysis that reveals not only shared curricular gaps but also differences in emphasis across institutional contexts. In doing so, it contributes new empirical evidence to the literature on Software Engineering education and supports future investigations on curriculum evolution, national standardization, and alignment with professional demands.

3 Research Method

The aim of this study is to analyze the landscape of Software Engineering degree programs offered by Brazilian higher education institutions, both public and private. It seeks to comprehensively understand how these courses are structured, identify their main characteristics, and assess the extent to which they align with current standards, guidelines, and norms. To address our primary objective, we formulated the following research questions:

- (RQ-1) How are Software Engineering degrees structured in public higher education institutions in Brazil?
- (RQ-2) How are Software Engineering degrees structured in private higher education institutions in Brazil?
- (RQ-3) To what extent do Brazilian undergraduate software engineering programs align with international curriculum guidelines?

The research method used in our study is presented in Figure 1, which is an adapted version of the *framework* by Zarour et al. (2025).

3.1 Data Selection

The first step consisted of extracting the list of courses from e-MEC[1], searching the documentation for each course, and analyzing the documents based on the information to be used in the analysis. e-MEC is a portal of the Brazilian Ministry of Education (MEC) used to monitor regulatory processes in Brazilian higher education. To obtain the list of courses, one of the authors of this article searched the e-MEC website for undergraduate Software Engineering courses offered in classroom mode at the bachelor's degree level and currently active. After the search, a detailed spreadsheet of the courses identified was exported, containing the information required for the analyses.

Based on the spreadsheet generated in e-MEC, two authors surveyed the documentation and other information for each course. The documents used in the analysis were the Course Pedagogical Project (PPC) or the curriculum matrices. To

[1] https://emec.mec.gov.br/emec/nova

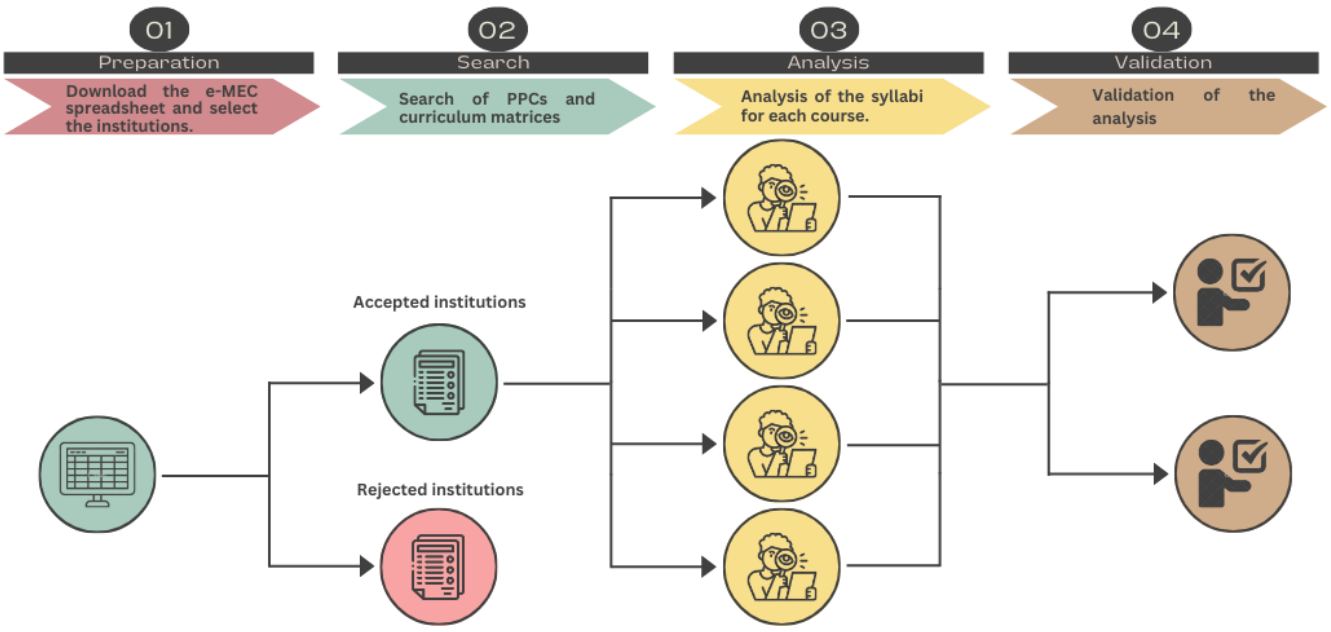


Figure 1. Research method

identify these documents, a search was conducted on the Web and on the institutions' websites, and the most recently available documents were selected. For private institutions, we took an additional step: contacting program coordinators directly when PPCs were not available on the institution's website. Additionally, filtering was performed based on inclusion and exclusion criteria. The inclusion criteria (IC) used were:

- IC 1 - Bachelor's degree courses offered in person;
- IC 2 - Courses starting by 2024.

The exclusion criteria (EC) used were:

- EC 1 - Courses starting after 2024;
- EC 2 - Courses that did not have syllabi available in their PPC or curriculum matrix. A syllabus is a descriptive document outlining the course plan for the discipline, including an overview of the topics to be covered and the scope of instruction.

3.2 Data Extraction and Analysis

After filtering, among the accepted institutions, four authors analyzed the documentation for each selected course and extracted the information necessary for analysis. The information collected was:

- **Name/acronym of the Institution**: Name or acronym of the Higher Education Institution;
- **Region**: Name of the Brazilian region where the institution is located;
- **State**: Name of the Brazilian state where the institution is located;
- **Municipality**: Name of the municipality where the institution is located;
- **Administrative category**: Whether it is a federal, state, or municipal public institution. Or whether it is private;
- **Course status**: Indicates whether the course has started or not;
- **Start year**: Year in which the course started;
- **Type of documentation**: Type of document analyzed (PPC or Curriculum Matrix);
- **Document year**: Year of publication of the most recent document;
- **Availability of syllabi**: Indicates whether the syllabi for the courses are available;
- **Required final project**: Indicates whether a final project is required;
- **Required internship**: Indicates whether a supervised internship is required.

In the analysis of course documentation that met the inclusion and exclusion criteria, four authors from two institutions were responsible for this stage. The authors analyzed aspects of the compulsory subjects offered in the courses, such as theoretical, practical, and extension workload, and the syllabi, seeking to align the planned teaching content with the training axes defined by the SBC and to assign each subject to an area of SWEBOK. The analysis, based on the educational axes defined by the SBC, was divided into two stages: first, the syllabus for each course was reviewed. After reading, the authors responsible for this stage mapped the planned content for each subject to the topics of each training axis, and the axis representing each subject was determined by the one with the most topics aligned with the subject syllabus. The analysis, based on the SWEBOK areas, involved mapping each subject to one of them. For this analysis, the area most closely aligned with the planned content for each subject was selected based on a review of the course syllabi. Each author analyzed course documentation from different regions of Brazil. The analyses performed by the authors were validated by other authors and by HE researchers working in HE teaching.

To complement the descriptive analysis with inferential statistics, two families of tests were applied depending on the nature of the comparison. For categorical distributions (e.g., the regional distribution of courses, and the distribution

of SWEBOK/SBC areas across regions or institution types) we used the chi-square test, both in its goodness-of-fit form (against a uniform distribution) and in its independence form (between two categorical variables), together with Cramér's V as a measure of effect size. For the comparison of per-course subject counts between public and private institutions within each area, where the data are discrete and the samples are small and unbalanced, we used the non-parametric Mann-Whitney U test, which does not require the normality assumptions of parametric procedures. All tests were two-tailed, with a significance level of $\alpha = 0.05$ and the null hypothesis of no difference between the groups compared. For the Mann-Whitney tests, effect size was additionally estimated with Cliff's δ (Romano et al., 2006), interpreted as negligible ($|\delta| < 0.147$), small ($|\delta| < 0.33$), medium ($|\delta| < 0.474$), or large (otherwise); positive values indicate higher subject counts in public institutions, and negative values indicate higher counts in private institutions.

To complement the analysis and investigate possible causes of discrepancies in the presence of Software Engineering disciplines in syllabi, we examined the profiles of the professors who teach the courses, seeking to identify their main areas of expertise and interest. Initially, we searched for lists of professors within the PPCs themselves and, when this information was unavailable, we looked to the HEIs' official websites. Then, for each professor identified, we consulted their curriculum on the Lattes Platform of the National Council for Scientific and Technological Development (CNPq).

For the profile classification, we primarily based it on the professional summaries provided by the professors themselves. When it was not possible to clearly identify the areas of interest from the summaries, we also analyzed their most recent publications. To verify whether the presence of professors with a Software Engineering profile correlates with a greater offering of courses in this area, we used Pearson's correlation coefficient at a significance level of 0.05, with a one-tailed test and the alternative hypothesis of a positive correlation. The hypotheses considered were:

- Null hypothesis (H0): The proportion of faculty members with a background in Software Engineering is not linearly related to the proportion of Software Engineering courses offered at the institutions studied.
- Alternative hypothesis (H1): A higher proportion of faculty members with a background in Software Engineering is associated with a higher proportion of Software Engineering courses in the institutions studied.

4 Results

In this section, we organize the results per research question.

4.1 (RQ-1) How are Software Engineering degrees structured in public higher education institutions in Brazil?

The search results on e-MEC returned 33 SE courses from public institutions. However, only 28 courses met the criteria due to the lack of syllabi in the PPC and the initiation of five courses after 2024 (as detailed in the exclusion criteria in Section 3). The analyzed documents were published between 2018 and 2024.

The 28 courses are distributed across Brazil's regions as follows (Figure 2): North (28.6%), Midwest (25%), Northeast and South (21.4% each), and Southeast (3.6%, a single course). A goodness-of-fit χ^2 test against a uniform distribution does not indicate a statistically significant imbalance ($\chi^2(4) = 5.21$, $p = 0.27$, Cramér's $V = 0.22$); the observed variation falls within the range expected by chance.

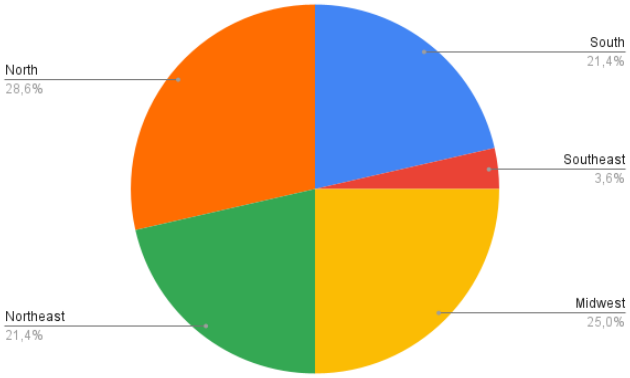


Figure 2. Distribution of courses across the regions of Brazil.

Figure 3 shows the distribution of public courses between capitals and the interior. Most regions concentrate courses in the interior; the Midwest is the exception, with most courses in the capital, while the Southeast has only one course, located in the interior. A chi-square test of independence between region and location was not statistically significant ($\chi^2(4) = 8.86$, $p = 0.065$, $V = 0.56$), so this pattern should be interpreted as descriptive rather than inferential.

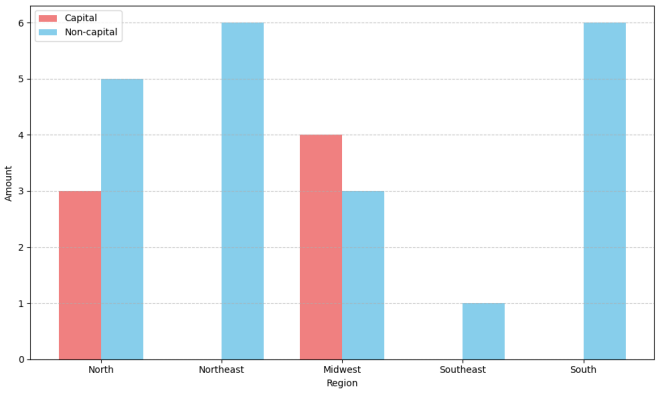


Figure 3. Distribution of courses across states and regions.

Figure 4 shows the cumulative evolution of Software Engineering courses at public institutions in Brazil from 2008 to 2024. The number of courses increased over time, with 15 opened in the first 10 years and 13 in the last 7 years, suggesting a recent acceleration in the expansion of specialized Software Engineering education in Brazil.

The relationship between the subjects in Software Engineering courses and the SBC training areas is shown in Figure 5. Area 1 (Fundamentals of Computing, Mathematics, and Production) has the highest proportion of subjects, with 42.2%, followed by Area 6 (Software Construction and Test-

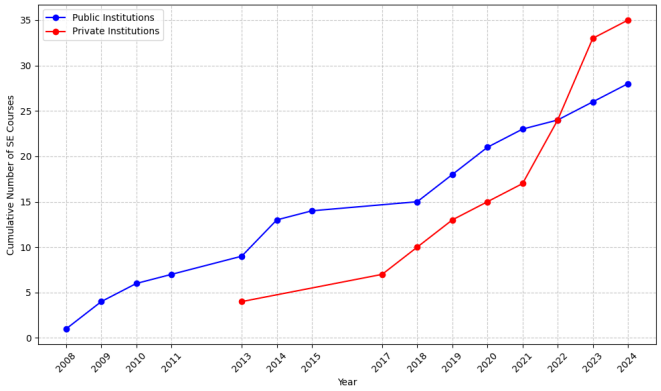


Figure 4. Evolution of the creation of Software Engineering undergraduate programs in public and private institutions in Brazil.

ing) with 19.8% and Axis 3 (Complementary Professional Skills and Practices) with 13.1%. In contrast, Axis 7 (Software Quality) is the least-addressed area in SE courses at public universities in Brazil, accounting for only 3.5%.

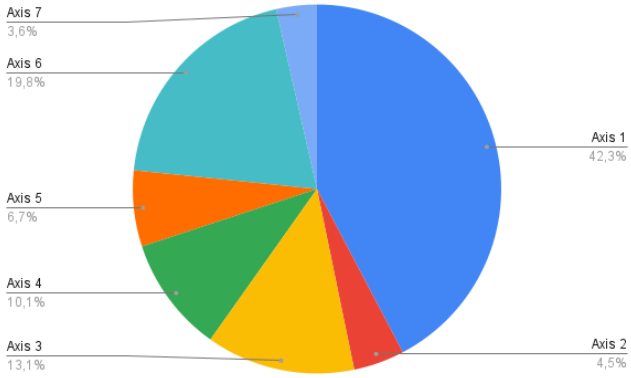


Figure 5. Adherence to the SBC's training axes in public institutions.

To analyze subject coverage across the SWEBOK areas, the SE courses were organized by Brazilian region, as illustrated in Figure 6. We excluded fundamental subjects, such as computing, mathematics, and engineering, to focus on areas more directly linked to Software Engineering. This decision enabled us to refine the analysis by focusing on areas of knowledge more directly linked to Software Engineering. However, the repository containing data from all courses in the regions is available in the artifact availability section.

In the North, the most frequent areas are "Software Engineering Management" and "Software Construction" (18.2% each), followed by "Professional Software Engineering Practice" (13.6%), while "Software Architecture" (1.3%), "Software Security" (1.9%), and "Software Requirements", "Software Quality", and "Software Configuration Management" (2.6% each) are the least represented.

In the Midwest, "Professional Software Engineering Practice" (22.8%), "Software Construction" (17.3%), and "Software Engineering Economics" (10.2%) account for the largest shares, whereas "Software Engineering Models and Methods" (0.8%), "Software Maintenance" (2.4%), and "Software Security" (2.4%) appear less frequently.

In the South, "Professional Software Engineering Practice" (26.5%), "Software Construction" (14.7%), and "Software Design" (9.8%) are the most prominent areas, while "Software Engineering Models and Methods" (1.0%), "Software Engineering Configuration Management" (2.0%), and "Software Maintenance" (2.0%) have the lowest coverage.

The Northeast shows a similar pattern, with "Professional Software Engineering Practice" (19.1%), "Software Construction" (15.7%), and "Software Design" (11.3%) leading the distribution, and "Software Engineering Models and Methods" (1.7%), "Software Maintenance" (3.5%), and "Software Security" (3.5%) among the least represented areas. The regional profiles above reflect descriptive patterns. A chi-square test of independence between SWEBOK area and region does not detect statistically significant variation ($\chi^2(52) = 50.77$, $p = 0.52$, $V = 0.16$), with the Southeast's single institution further limiting regional comparisons.

The graph in Figure 7 shows the average number of subjects per course by SWEBOK area. This graph shows the sum of all subjects across all areas divided by the number of courses analyzed. The areas of "Professional Software Engineering Practice", "Software Construction", "Software Engineering Economics", "Software Engineering Management", "Software Design", and "Software Engineering Processes" exceed the descriptive reference of one subject per program.

In contrast, the areas of "Software Quality", "Software Testing", and "Software Architecture" are below the reference line. "Software Maintenance" and "Software Engineering Models and Methods" represent less than half a course each in the SE programs of the public institutions analyzed, indicating a low presence in these programs.

Regarding Software Quality, 10 educational institutions offer no specific courses. In the area of Software Requirements, seven educational institutions offer no courses related to it. Finally, in Software Testing, only three institutions do not offer this course.

Of the 28 institutions, only 17 had faculty lists available. The analysis of public institutions revealed a moderate positive correlation between the proportion of faculty with a Software Engineering (SE) background and the proportion of SE-related courses offered. The Pearson correlation coefficient was r=0.5162 with a statistically significant p-value (p=0.0339<0.05) under a one-tailed test, allowing the rejection of the null hypothesis (H0) in favor of the alternative hypothesis (H1).

Overall, institutions with a higher proportion of SE-oriented faculty tended to offer a greater share of SE disciplines. For instance, IFSP presented the highest proportion of SE faculty (55.6%) and also one of the highest proportions of SE courses (57.1%). Similarly, IFG (45.5% faculty; 51.2% courses), IFPR (44.4% faculty; 51.6% courses), and UFR (45.5% faculty; 53.8% courses) followed this trend. Conversely, institutions with lower proportions of SE faculty, such as UEPG (5.0% faculty) and UFAM Manaus (11.8% faculty), exhibited comparatively lower proportions of SE courses (39.0% and 45.7%, respectively).

However, some deviations from this pattern were observed. For example, IFAM Parintins, despite having only 12.5% of faculty with an SE background, offered a relatively high proportion of SE courses (53.8%). Similarly, UTFPR Cornélio Procópio showed a moderate proportion of faculty (21.6%) but a high proportion of SE disciplines (56.8%). These cases suggest that additional institutional or curricular factors may also influence the distribution of SE content.

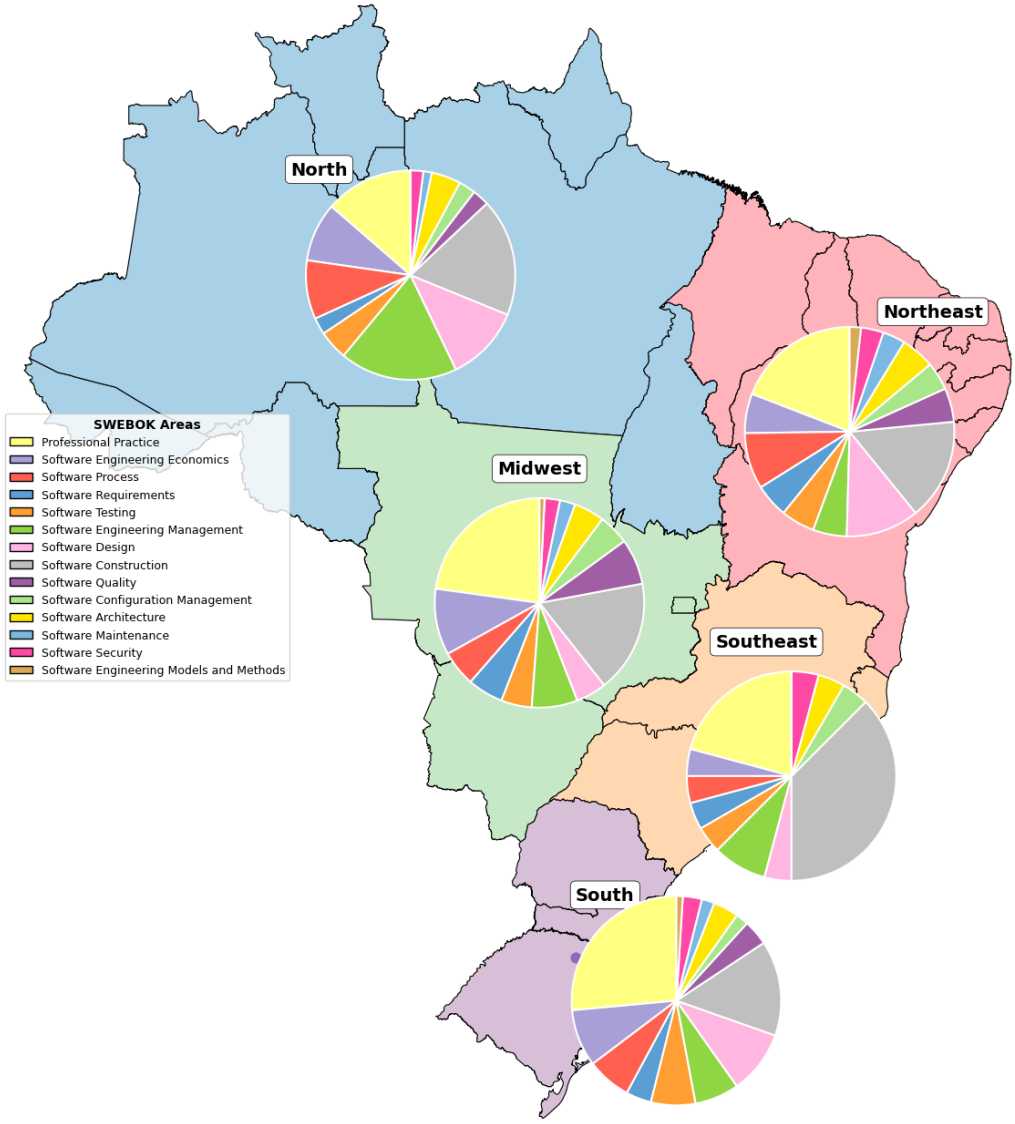


Figure 6. Distribution of SWEBOK areas by region in public institutions.

4.2 (RQ-2) How are Software Engineering degrees structured in private higher education institutions in Brazil?

Based on a search of the e-MEC system, we identified 141 undergraduate Software Engineering courses at private institutions. Of these, 24 were listed as not yet started. When syllabi were unavailable in the PPCs or on institutional websites, we requested them directly from course coordinators via email. Courses were excluded only when coordinators did not respond or explicitly stated that they could not provide the syllabi. Under these conditions, 82 courses were excluded, resulting in a final sample of 35 courses.

Figure 10 shows that private Software Engineering courses are unevenly distributed across Brazil. The South and Southeast together account for 80% of the courses (42.9% and 37.1%, respectively), followed by the Midwest (8.6%), while the Northeast and North each account for 5.7%. This concentration is statistically confirmed: a goodness-of-fit test rejects uniform regional distribution for private institutions ($\chi^2(4) = 23.71$, $p < 0.001$, $V = 0.41$), and a test of independence between institution type and region confirms that public and private distributions differ significantly ($\chi^2(4) = 20.82$, $p < 0.001$, $V = 0.58$).

Figure 4 shows the annual distribution of course creation in private institutions. The data indicate the creation of 35 courses, with the first initiatives recorded from 2013 onwards. Since then, there has been a more pronounced increase in the number of courses opened, especially between 2020 and 2023, a period during which a significant portion of the new offerings are concentrated.

Figure 8 compares the proportion of private courses located in capital cities and in the interior. Overall, most courses are located in the interior of the states. This pattern is especially pronounced in the South and Southeast, where more than 80% of the courses are located outside capital cities. In contrast, the Midwest, Northeast, and North show the opposite pattern, with most courses concentrated in capitals. In the latter two regions, all identified courses are located in capital cities. A chi-square test of independence between region and location confirms a statistically significant association for private institutions ($\chi^2(4) = 12.92$, $p = 0.012$, $V = 0.61$).

Figure 9 illustrates the alignment between subjects in pri-

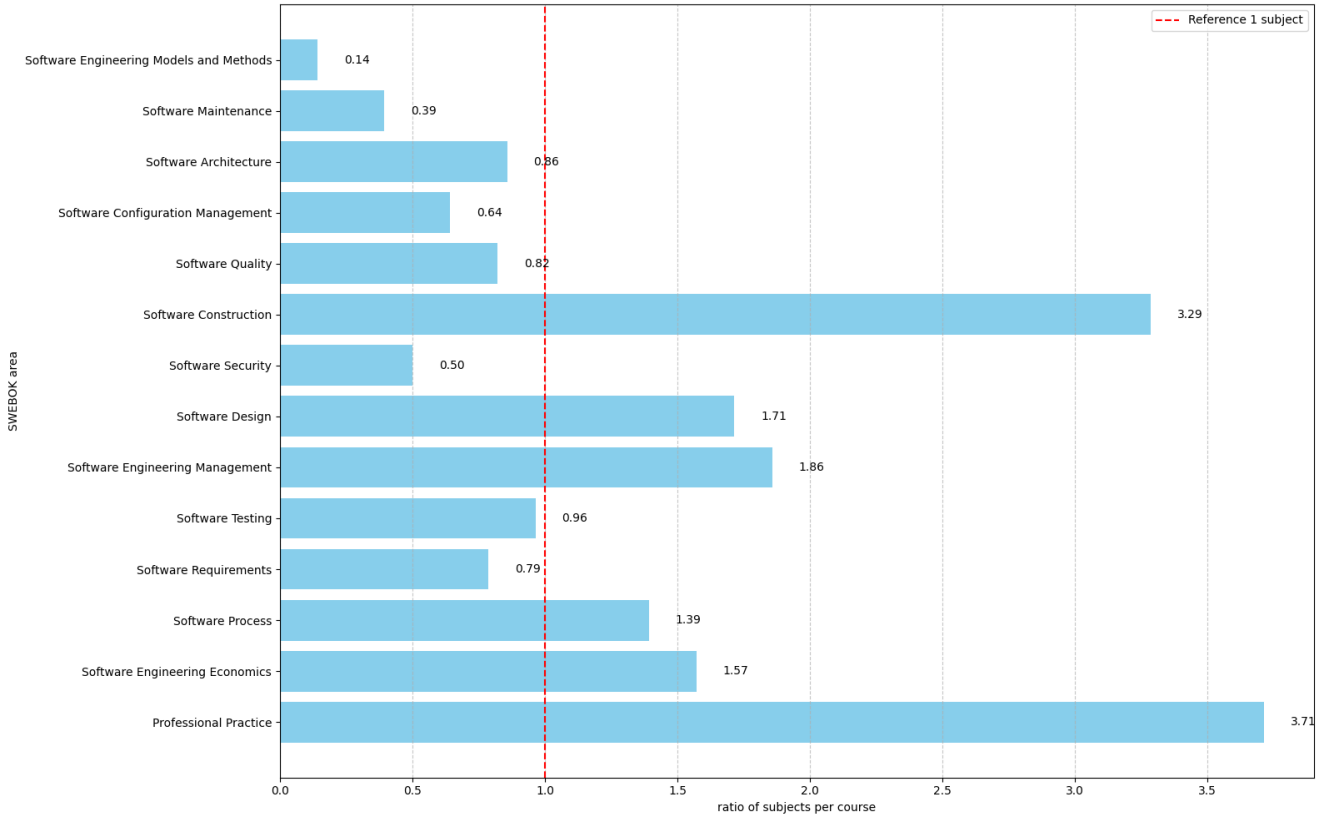


Figure 7. Average number of subjects per course by SWEBOK area, based on public courses in Brazil

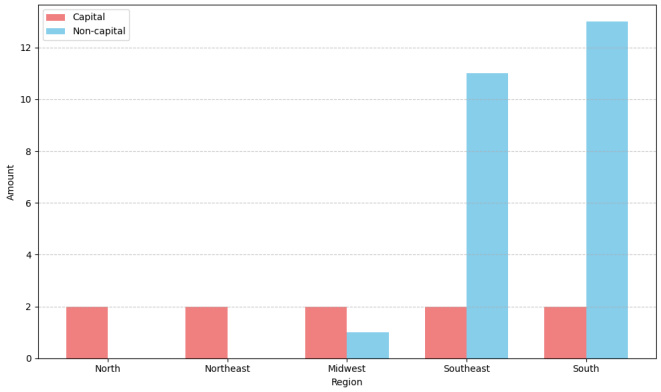


Figure 8. Distribution of private courses across states and regions.

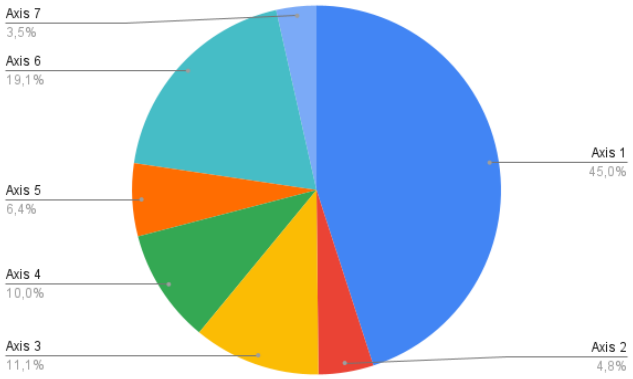


Figure 9. Adherence to the SBC's training axes in private institutions.

vate Software Engineering courses and the SBC training axes. Axis 1 (Computing Foundations, Mathematics, and Production) has the largest share of subjects (45.0%), followed by Axis 6 (Software Construction and Testing) with 19.1% and Axis 3 (Complementary Professional Skills and Practices) with 11.1%. In contrast, Axis 7 (Software Quality) is the least represented, accounting for only 3.5%.

The analysis of subject coverage across SWEBOK areas was organized by region, as shown in Figure 11. As in the analysis of public institutions, subjects related to the fundamentals of computing, engineering, and mathematics were excluded. However, the repository with data on all subjects from the regions is available on Zenodo.

In the South, which contains the largest number of analyzed Software Engineering courses, the most prevalent SWEBOK areas are "Professional Practice of Software Engineering" (27.7%), "Software Construction" (15.5%), and "Software Engineering Management." The least represented are "Software Engineering Models and Methods" (1.1%), "Software Engineering Operations" (1.5%), and "Software Configuration Management" (2.7%).

A similar pattern appears in the Southeast, where "Professional Practice of Software Engineering" (26.9%), "Software Construction" (19.8%), and "Software Engineering Management" (9.9%) account for the largest shares. The least represented areas are "Software Configuration Management" (1.6%), "Software Maintenance" (2.2%), and "Software Testing" (2.7%).

In the Midwest, the most frequent areas are "Professional Practice of Software Engineering" (30.4%), "Software Construction" (17.4%), and "Software Engineering Processes" (8.7%). By contrast, "Software Engineering Models and Methods" and "Software Configuration Management" are the least frequent (2.2% each), followed by "Software Engi-

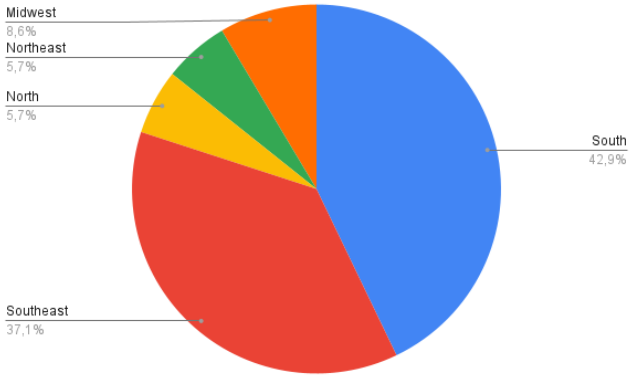


Figure 10. Distribution of private courses across the regions of Brazil.

neering Management" and "Software Testing" (4.3% each).

The Northeast region shows, among the most represented areas, "Professional practice of software engineering" (33.3%), "Software Design" (14.8%), and "Software engineering process" (11.1%). In contrast, 7 areas are tied with 3.7%, among which "Software Quality", "Software Requirements", and "Software Maintenance" stand out.

Finally, in the North region, the areas with the highest representation are "Professional Practice of Software Engineering" (26.8%), "Software Design" (12.2%), and "Software Engineering Economics" and "Software Construction" (tied at 9.8%). On the other hand, the areas with the fewest disciplines are "Software Quality" and "Software Architecture," both with 2.4%. As with public institutions, a chi-square test of independence between SWEBOK area and region for private institutions is not significant ($\chi^2(56) = 42.47$, $p = 0.91$, $V = 0.14$); these regional profiles are descriptive patterns, not statistically confirmed differences.

Figure 12 shows the average number of subjects per program in each SWEBOK area. The areas of "Software Construction," "Software Design," "Software Engineering Management," "Software Engineering Processes," and "Professional Practice of Software Engineering" each exceed the descriptive reference of one subject per program. By contrast, "Software Engineering Economics," "Software Architecture," and "Software Requirements" fall below this reference line, while "Software Maintenance," "Software Configuration Management," and "Software Engineering Models and Methods" average fewer than 0.5 subjects per program.

When comparing software engineering courses offered in public and private institutions, we see that both prioritize "Professional Practice" (3.71 in public institutions and 4.71 in private institutions), indicating an emphasis on professional skills in both modalities; "Software Construction" (3.29 in public institutions and 3.06 in private institutions), showing a focus on practical implementation; and, finally, management and process areas with an average score above 1 (such as "Software Engineering Management", with 1.86 in public institutions and 1.66 in private institutions).

The software engineering courses offered in both modalities also present shared curricular gaps, such as low rates in "Software Quality" (0.82 in public institutions and 0.66 in private institutions); "Software Testing" (0.96 in public institutions and 0.6 in private institutions), and "Software Architecture" (0.86 in public institutions and 0.80 in private institutions), all below 1, indicating curricular gaps.

On the other hand, the descriptive averages suggest a difference in emphasis between the two sectors. Courses offered in public higher education appear to strike a better balance between theoretical and support areas, whereas private institutions focus more on immediate practice. This impression is consistent with the per-course means, which favor public institutions in the economic aspects of software (1.57 versus 1.03), "Software Design" (1.71 versus 1.26), "Software Processes" (1.39 versus 1.26), and "Software Testing" (0.96 versus 0.60), and favor private institutions in "Professional Practice" (4.71 versus 3.71). Public institutions also have a higher average across areas (1.32 versus 1.26), suggesting broader curricula. Most of these descriptive differences are not supported by inferential testing. Two-sided Mann-Whitney U tests showed no statistically significant difference between the sectors in "Professional Practice" ($p = 0.066$, Cliff's $\delta = +0.27$), "Software Design" ($p = 0.086$, $\delta = +0.24$) or "Software Processes" ($p = 0.232$, $\delta = +0.16$). Only "Software Engineering Economics" ($p = 0.027$, $\delta = +0.30$) and "Software Testing" ($p = 0.008$, $\delta = +0.33$) presented significant difference.

Of the 35 institutions, only 16 had faculty lists available. The analysis did not reveal a statistically significant correlation between the proportion of faculty with an SE background and the proportion of SE-related courses offered. The Pearson correlation coefficient was r = 0.0413, with a non-significant p-value (p = 0.8794 > 0.05) under a one-tailed test. Therefore, we fail to reject the null hypothesis (H0), indicating no evidence of a positive linear relationship between these variables in the context of private institutions.

In general, the data exhibit high variability, with no clear pattern linking faculty background to curricular offerings. For example, FADIC showed a relatively high proportion of SE faculty (40.0%) alongside a high proportion of SE courses (65.7%). In contrast, UNIFIO, despite having an even higher proportion of SE faculty (50.0%), presented a comparatively lower proportion of SE disciplines (40.6%). Similarly, institutions such as UBM (7.7%) and Univassouras Saquarema (14.3%) had low proportions of SE-oriented faculty but offered high proportions of SE courses (61.1% and 58.5%, respectively).

Conversely, some institutions with low proportions of SE faculty also exhibited low or moderate curricular presence, such as UNILINS (5.0%; 37.3%) and UNIFATECIE Paranavaí (5.3% faculty; 42.9% courses), reinforcing the absence of a consistent trend. These mixed patterns suggest that, in private institutions, factors other than faculty specialization, such as market-driven curriculum design, institutional policies, or standardized course structures, may play a more significant role in shaping the presence of Software Engineering disciplines.

(RQ-3) To what extent do Brazilian undergraduate software engineering programs align with international curriculum guidelines?

Based on the analysis of the international curricula presented in Table 1, Figure 13 compares the proportions of subjects across Brazilian regions and Brazil as a whole against the reference knowledge areas adopted in international curricula. Because these knowledge areas overlap with SWE-

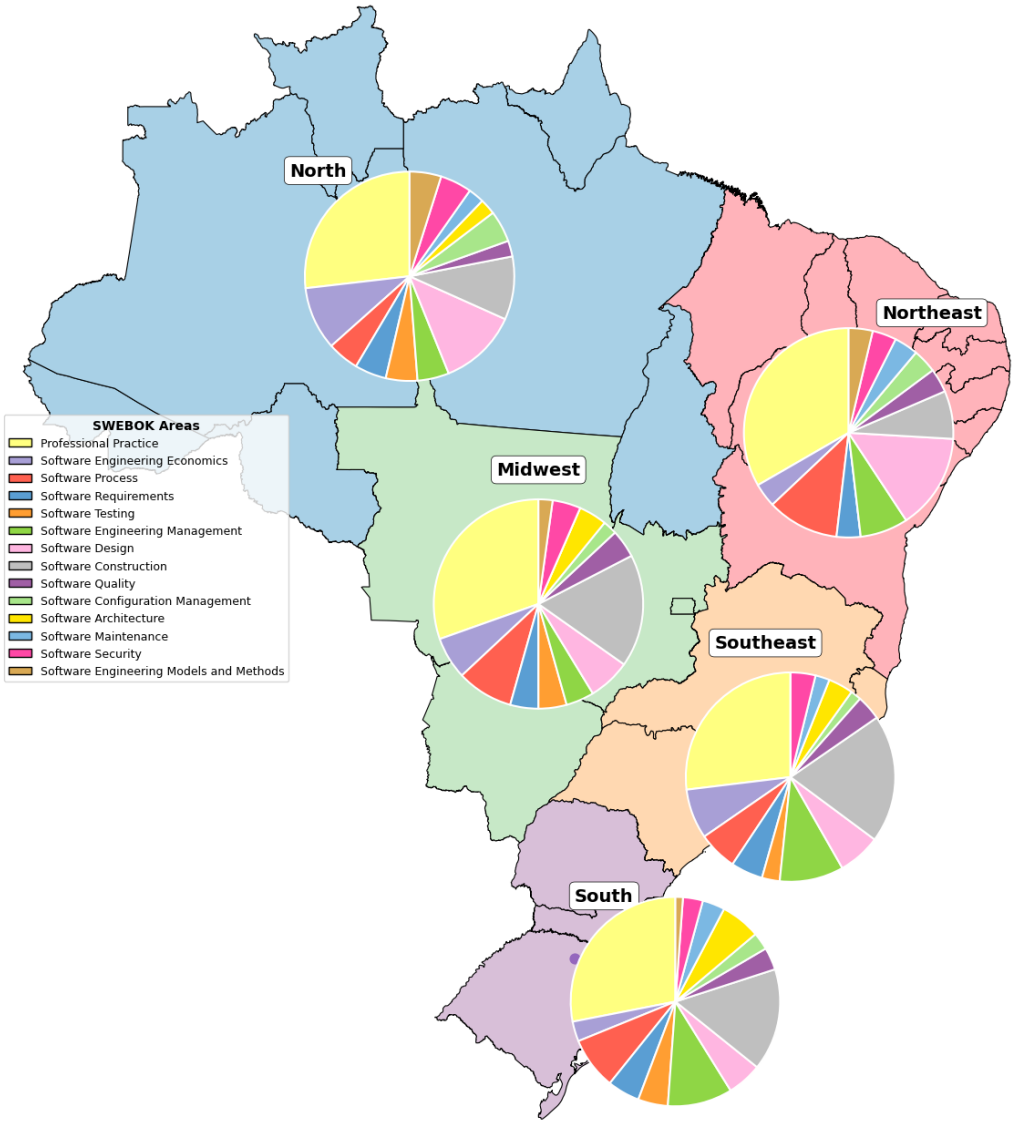


Figure 11. Distribution of SWEBOK areas by region in private institutions.

BOK areas, the percentages reported by region and for Brazil may sum to more than 100%.

In public institutions, “Fundamentals of Computer Science and Programming” is the most representative area, especially in the Southeast (57.1%), North (47.2%), and South (47.0%). These values are substantially higher than those observed in international curricula such as India (28.0%) and Russia (23.2%). “Software Engineering Core” also has a strong presence in Brazil, particularly in the Northeast (25.8%), South (20.3%), and Midwest (19.8%), whereas it appears less prominently in international curricula and is absent from the Indian curriculum. By contrast, “Advanced Topics and Fundamentals of Engineering” is more prominent in international curricula, especially in India (48.5%) and Germany (33.3%), than in Brazilian public curricula, where the highest proportions are found in the Midwest (16.2%) and South (15.4%).

A similar pattern appears in private institutions. ”Fundamentals of Computer Science and Programming” is the most representative subject in curricula, particularly in the Southeast (47.2%), South (46.2%), and Midwest (42.9%) regions. These percentages surpass those found in international curricula, such as those of India (28%) and Germany (24.5%). Following this, the ”Software Engineering Core” also shows a strong presence in Brazil, especially in the Midwest (32.1%), South (30.1%), and North (29.6%) regions. At the same time, in international curricula, this area appears less prominently or is even absent, as in the Indian curriculum (0%). In contrast, ”Advanced Topics and Fundamentals of Engineering” are more frequently addressed in international curricula, particularly in India (48.5%) and Germany (33.3%), than in Brazilian curricula, where the highest percentages are observed in the Northeast (25%) and Midwest (21.4%).

For “Mathematical Foundations,” international curricula again show higher proportions, particularly in Russia (18.9%) and Germany (15.4%). In Brazil, the highest values among public institutions are in the Northeast (14.6%) and South (9.8%), with a national average of 10.0%, while among private institutions, the highest values are in the Northeast (11.5%) and Southeast (10.4%), with a national average of 8.9%. By comparison, “Economics, Ethics and Management” appears more strongly in Brazilian curricula, espe-

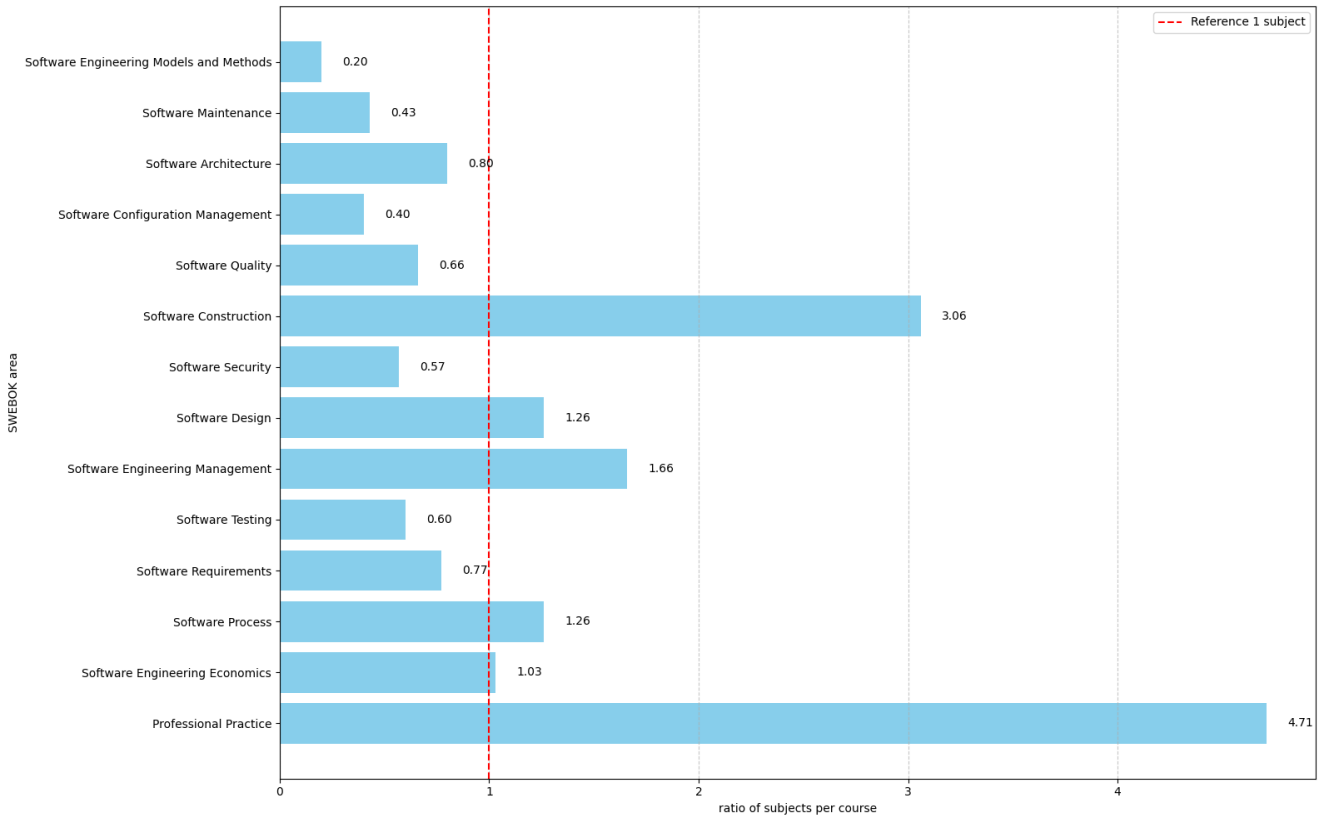


Figure 12. Average number of subjects per course by SWEBOK area, based on private courses in Brazil.

cially in public institutions in the South (20.1%) and Southeast (19.1%), with a national average of 18.2%, and in private institutions in the Northeast (23.1%), with a national average of 21.0%. These values exceed those found in international curricula, which range from 0% in Germany to 7.5% in the ACM curriculum. Finally, "Technical Writing and Presentation" is relatively well distributed across Brazilian regions, with higher proportions in the South (12.6%) and Southeast (11.9%) among public institutions, and in the Northeast (17.3%) among private institutions, compared with more modest proportions in international curricula, such as Germany (2.5%) and India (3.0%).

When comparing the curricula of higher education programs in Brazilian public and private institutions in the analyzed areas, we observed that both prioritize "Fundamentals of Computer Science and Programming," with similar national averages (45% in private institutions and 46% in public institutions), although public institutions stand out more in the Southeast region (57.1%) and private institutions in the Northeast region (47.2%). In the "Software Engineering Core," private institutions show greater representation (average 26.2%, peaking in the Central-West at 32.1%) than public institutions (average 21.1%, with a strong presence in the Northeast at 25.8%), suggesting a greater practical emphasis in private institutions. In "Fundamentals of Engineering," private institutions lead with an average of 19% (Northeast 25%) against 14.2% in public institutions (Central-West 16.2%), widening the difference in this technical dimension. In "Economics, Ethics and Management," private institutions cover more subjects (average 21%, Northeast 23.1%) than public institutions (18.2%, South 20.1%). In comparison, "Mathematical Foundations" is slightly higher in public institutions (average 10%, Northeast 14.6%) compared to private ones (8.9%, Northeast 11.5%). Finally, "Technical Writing" favors private institutions (average 13.74%, Northeast 17.3%) over public ones (9.5%, South 12.6%), highlighting complementary patterns between the two types of institutions. These comparisons describe the proportional distribution of broad curriculum categories under the mapping adopted in this study. They should not be interpreted as direct evidence of differences in instructional quality or curricular depth between institutional sectors.

5 Discussion

Based on data from 2005 analyzed in da C. Figueiredo et al. (2010), it was found that there was a need to create undergraduate courses in Software Engineering in Brazil to meet the growing demand from the national software industry. Considering what was presented in Figure 4, this need has been gradually met with the creation of undergraduate courses in SE at public institutions and a gradual increase over the years. Approximately 28 undergraduate courses in SE emerged between 2008 and 2024, highlighting both the incentive for training and the increased demand for Software Engineering courses in public education in Brazil.

However, the data do not provide statistically significant evidence that the regional distribution of public programs differs from a uniform distribution, as shown in Figure 2. This distribution shows that regions such as the North and Midwest have the highest number of courses in public institutions. Meanwhile, the Southeast has the fewest courses in

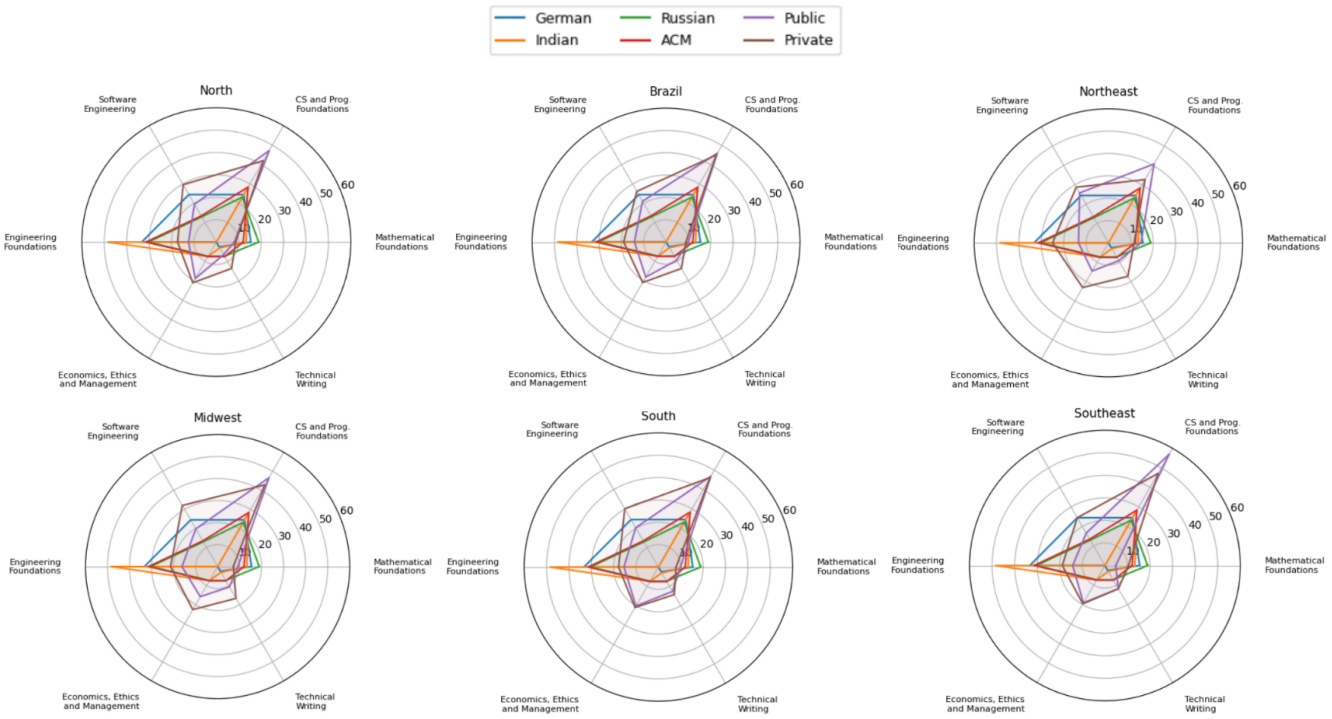


Figure 13. Comparison of curricula by knowledge area and region.

the country, with only one course offered at a public institution. Descriptively, most SE courses in public institutions are located in the interior of the states, which may suggest an incentive to bring higher education to non-capital cities; however, this pattern does not reach statistical significance (see Section 3). Whether state capitals with strong software industries are systematically underserved remains an open question warranting further investigation.

With regard to the analysis of the SBC curriculum axes, Figure 5, it is clear that almost half of the subjects taught in SE courses are related to Axis 1 (Fundamentals of Computing, Mathematics, and Production), which indicates an emphasis on training based on these fundamentals and shows that there is not the same adherence to the other axes.

Ferreira et al. (2018) point out emerging topics in SE (such as experimental software engineering, reuse, refactoring, and software testing techniques) and relates them to the SBC training axes. The axes related to emerging topics were 1, 3, 4, 5, 6, and 7. Despite the emergence of these topics, most of the axes mentioned are not represented in the curricula of SE courses, particularly axes 5 (Requirements, Analysis, and Software Design) and 7 (Software Quality). Given that these axes encompass important knowledge for the elicitation, analysis, validation, and management of software requirements and software architecture, as well as techniques for ensuring product and development process quality, their absence may indicate a gap in the training of future professionals in Software Engineering.

An analysis of software engineering courses offered by private institutions reveals a significant expansion of this type of education in Brazil, though regional and curricular disparities accompany this expansion. The strong concentration of courses in the south and southeast regions suggests that the expansion of private SE education follows the historical distribution of the country's main economic and technological hubs. This pattern suggests that the opening of private courses may be associated with regional demand for software professionals and the presence of more consolidated technological ecosystems.

Another relevant aspect concerns the geographical distribution of courses between capital cities and the interior. In the South and Southeast regions, there is a significant presence of courses located outside the capital cities, indicating that the private sector plays an important role in bringing software engineering training to the interior of these regions. In contrast, in the North and Northeast regions, the courses identified are concentrated in the capitals, which may reflect structural limitations or a lower density of private technology-specializing institutions in the interior of these regions. This scenario shows that although private education contributes to expanding the offer of SE courses nationwide, their territorial distribution still reflects regional inequalities.

Regarding curriculum organization, the predominance of Axis 1 of the SBC guidelines indicates that private courses maintain a training base strongly grounded in the fundamentals of computing, mathematics, and production. This result suggests that part of the curricula still preserves a strong influence from traditional training in computing courses, such as Computer Engineering and Computer Science. At the same time, the significant presence of subjects related to Software Construction and Professional Practices indicates an attempt to reconcile conceptual foundations with practical software development.

However, the low presence of subjects related to Software Quality reveals a possible gap in the training offered by these institutions. The analysis of the SWEBOK areas reinforces this observation, indicating that key areas such as Software Maintenance, Software Quality, Software Security, Software Architecture, Configuration Management, and Software Engineering Models and Methods are underrepresented in the

analyzed curricula. Given that these areas play a fundamental role in the maintenance and evolution of software systems over time, their limited presence may indicate that certain dimensions of software engineering still receive less attention in the curricula of some private courses.

Compared to courses offered by public institutions, the curricula of private institutions have a higher concentration of subjects in Professional Practices, with an average of 4.71 subjects per course, while in public institutions this average is 3.71. This result suggests that private institutions place greater emphasis on skills related to professional practice and preparation for activities directly associated with the job market. This difference, however, did not reach statistical significance under a two-sided test, so it should be interpreted as a consistent tendency, supported by a small effect size, rather than a confirmed gap.

On the other hand, courses offered by public institutions have higher averages in several core areas of Software Engineering, such as Software Construction, Software Design, Software Engineering Management, Software Engineering Economics, Software Engineering Process, and Software Testing. This pattern indicates a more balanced distribution of curriculum across the different dimensions of software engineering. It should be noted, nonetheless, that when these area-by-area differences are tested individually with two-sided non-parametric tests, only "Software Engineering Economics" and "Software Testing" remained statistically significant.

Finally, the analysis of the relationship between faculty background and curricular composition reveals distinct dynamics between public and private institutions. In public institutions, a moderate, statistically significant positive correlation ($r=0.5162$, $p<0.05$) indicates that faculty with a Software Engineering background are associated with a greater proportion of SE-related courses. This finding indicates that, in the public context, faculty expertise may play a relevant role in shaping curricular decisions, possibly reflecting greater academic autonomy or alignment between research interests and teaching activities.

In contrast, private institutions showed no significant correlation ($r = 0.0413$; $p > 0.05$), indicating that the proportion of SE faculty is not associated with the proportion of SE disciplines offered. The absence of a statistically significant association indicates that faculty background alone does not explain the curricular composition of the analyzed private programs. Other organizational and contextual factors may be relevant, but their influence was not directly examined in this study.

Taken together, the findings are compatible with the possibility that curricular composition is shaped by different combinations of factors across institutional sectors. In public programs, faculty background was associated with the proportion of SE-related subjects, whereas no such association was observed in the analyzed private programs. This contrast suggests that faculty expertise may be one relevant factor in public institutions, while other organizational or contextual factors may be more influential in private institutions. However, these mechanisms were not directly examined and should be investigated through qualitative studies with program coordinators and institutional decision-makers.

5.1 Implications

These findings carry important implications for educational policy and curriculum governance. The persistent underrepresentation of areas such as Software Quality, Requirements Engineering, and Software Testing in both public and private institutions suggests that the current reference curriculum from the SBC, while providing a useful framework, may not exert sufficient normative influence on course design. This points to the need for more active mechanisms of curriculum oversight, such as periodic accreditation reviews that explicitly evaluate adherence to SWEBOK and SBC guidelines. Although the regional SWEBOK comparisons did not identify statistically significant differences, the limited number of analyzed private programs in the North and Northeast motivates further investigation into regional access, documentation availability, institutional capacity, and curriculum development.

Private programs presented a higher sample average in Professional Practice, although the difference did not reach statistical significance. This pattern motivates a hypothesis for future investigation: some private institutions may place greater emphasis on employability-oriented and practice-based components. Because curriculum decision-making and employer influence were not directly examined, this interpretation should be regarded as a plausible explanation rather than an empirically established mechanism.

Beyond these structural concerns, the curricular patterns observed in public and private institutions suggest inertia, inherited from traditional Computer Science and Computer Engineering courses. Even in programs designed as Software Engineering courses, the differentiation from general computer science curricula remains incomplete in practice, with foundational subjects predominating at the expense of core Software Engineering areas. This is evident in the persistent underrepresentation of Software Requirements, one of the most valued competencies in the industry according to Software Engineering literature, whose absence in formal curricula may contribute to the difficulties graduates face when entering complex, real-world software projects.

The regional dimension also deserves attention beyond institutional comparisons. The geographic concentration of private programs in more economically developed regions suggests that access to certain types of education and training is conditioned by regional development, rather than by a deliberate educational policy. This demands coordinated action among regulatory bodies, academic associations, and industry actors to ensure a more equitable distribution of comprehensive education and training across the country.

Furthermore, it is important to recognize that the curriculum structure, while allowing for the analysis of educational quality, is not sufficient on its own. The presence of a subject in the curriculum does not guarantee the depth, rigor, or pedagogical approach with which it will be taught. Future investigations should therefore complement structural analyses such as this one with qualitative data on teaching practices, faculty expertise, and student outcomes, including employability indicators, to offer a more complete understanding of the quality of higher education in Brazil.

For program coordinators, the findings provide a diagnos-

tic basis for reviewing whether core SE areas are addressed with sufficient visibility, continuity, and integration across the curriculum. Such reviews should examine not only the existence of dedicated subjects but also the distribution of learning outcomes across compulsory and elective components. For policymakers and professional associations, the recurrent patterns identified across institutional sectors indicate the value of periodic national curriculum monitoring and guidance mechanisms that promote comparability without suppressing institutional autonomy. For researchers, the results could be applied to define a research agenda involving curriculum decision-making, classroom practices, faculty expertise, student learning, graduate employability, and longitudinal curriculum evolution. Industry partners can also contribute through advisory boards, real projects, internships, and collaborative curriculum reviews, provided that immediate employability demands are balanced with the long-term engineering competencies expected from Software Engineering graduates.

5.2 Curricula comparison

A comparison of Brazilian software engineering curricula with international curricula from India, Russia, and Germany reveals some differences in the distribution of training areas. In general, Brazilian courses devote a larger portion of the curriculum to computer science fundamentals and programming. This pattern appears in both public and private institutions, indicating that the Brazilian model, in both administrative spheres, devotes a larger portion of the curriculum to computer science and programming than the international standards analyzed.

By comparison with the international curricula analyzed, Brazilian courses place less emphasis on advanced engineering topics. In countries such as India and Germany, these topics occupy a larger share of the curriculum, suggesting a greater focus on specialized aspects of software engineering. This scenario suggests that, although Brazil has a well-representative foundation, it lacks advanced engineering content relative to the international curricula of India and Germany.

Another point of divergence refers to the presence of the specific core of Software Engineering. While this dimension appears in a limited form or is absent in some international curricula analyzed, it is well represented in Brazilian courses, especially in private institutions. This result suggests that courses in Brazil seek to explicitly structure a set of disciplines focused on the practices and processes characteristic of Software Engineering.

In addition, it is observed that complementary training dimensions, such as management, professional ethics, technical writing, and communication, appear with greater frequency in Brazilian curricula than in some of the international references analyzed. These elements indicate a concern with developing professional competencies beyond technical development skills.

The higher presence of computing and programming foundations in Brazilian curricula should not be interpreted solely as evidence of curricular deficiency. This configuration may partially reflect contextual characteristics of Brazilian and Latin American Software Industries, in which implementation-oriented roles constitute an important entry point for graduates and employers frequently demand immediately applicable programming skills. According to the Brazilian Software Market (ABES) report[2], "The Brazilian IT market registered growth of 13.9% in 2024, surpassing the global average of 10.8%, consolidating Brazil as the largest IT player in Latin America, responsible for 34.7% of investments in the region". It may also respond to differences in students' prior preparation, leading institutions to dedicate a substantial portion of the curriculum to foundational computing competencies.

A programming foundation is relevant for Software Engineering education. However, it may become restrictive when it systematically reduces formal exposure to areas required for the development and evolution of complex software-intensive systems, including architecture, requirements, quality assurance, testing, maintenance, security, and operations.

Our analysis does not allow us to determine whether the observed curricular distribution results from deliberate adaptation to Software Industry demands, historical inheritance from Computer Science programs, institutional resource constraints, regulatory incentives, or faculty availability. Accordingly, we interpret the Brazilian pattern as a curricular tension between immediate technical preparation and comprehensive engineering education, rather than as evidence that the national model is intrinsically inferior to the international references considered. Future qualitative studies involving program coordinators, faculty members, employers, students, and graduates are required to examine the mechanisms underlying these curricular decisions.

6 Threats to the validity

This section discusses the main threats to the validity of this study, organized according to the categories proposed for empirical studies in Software Engineering (Wohlin et al., 2012).

Construct Validity. The primary instrument of analysis was the course syllabus (in Portuguese, *ementa*), which describes the planned content of each discipline. However, a syllabus is a formal document and may not fully capture the depth, emphasis, or pedagogical approach with which topics are actually delivered in the classroom. A course may list "Software Testing" as a topic within a broader discipline without dedicating sufficient instructional time to it. Conversely, some instructors may cover relevant content beyond what is formally documented. This limitation is inherent to curriculum analysis studies and has been acknowledged in similar works (Zarour et al., 2025). To partially mitigate this threat, our analysis considered the full syllabus of each discipline, not only its title, seeking to identify the topics actually planned for instruction. The identification of faculty members with a Software Engineering profile was based on information from course syllabi, institutional websites, and Lattes curricula, prioritizing professional summaries and, when necessary, recent publications. This operationalization may introduce classification errors, especially in hybrid profiles,

[2] abes.org.br/dados-do-setor

affecting the accuracy of the proportion of Software Engineering faculty members per course.

A second construct validity concern relates to the mapping of disciplines to SWEBOK areas and SBC axes. Both reference frameworks encompass broad, sometimes overlapping knowledge areas, and mapping a single discipline to a single area requires interpretive judgment. In cases where a discipline's syllabus aligned with multiple areas, the authors selected the area with the greatest topical overlap. To reduce subjectivity, all mappings were independently reviewed by at least one additional author, and disagreements were resolved through discussion. Nonetheless, some degree of classification ambiguity cannot be fully eliminated.

Internal Validity. A primary concern in this study is analyst bias in interpreting course syllabi. The four authors responsible for the analysis were from two different institutions and analyzed documentation from different Brazilian regions, thereby reducing the risk of regional bias. All analyses were cross-validated by other members of the research team, including researchers with experience in higher education teaching. However, because the analysis is qualitative and relies on human judgment, some subjectivity in the classification process cannot be entirely ruled out. In analyzing the faculty profile, we correlated the current faculty snapshot with the subjects listed in the syllabus, without controlling for confounding variables (such as institutional policies and the course's history). Thus, it is not possible to establish causality. Only a statistical association between the faculty profile and the proportion of SE-related subjects can be established. Furthermore, both the course syllabi and the Lattes curricula and institutional websites may be outdated, leading to a temporal misalignment between the measured profile and the current reality.

External Validity. Regarding public institutions, the sample of 28 courses comprises all active public SE undergraduate programs in Brazil that met the inclusion criteria, thereby strengthening the representativeness of the findings for this segment. However, for private institutions, the generalizability of findings is more limited. Of the 141 private SE courses identified in e-MEC, 24 had not yet started, and 82 were excluded due to the unavailability of syllabi, even after direct email contact with course coordinators. The final sample of 35 private courses, therefore, represents approximately 30% of active private SE programs in Brazil. Courses whose coordinators did not respond or declined to share documentation may differ systematically from those included in the analysis. This potential selection bias should be considered when interpreting the findings related to private institutions.

Conclusion Validity. In this study, inferential statistical tests were applied to the main regional and institutional comparisons (see Section 4). However, the reduced sample size and the potential selection bias in the private sample (see External Validity) still may affect statistical power, since the tests assume that the analyzed courses are representative of their underlying populations. Per-area comparisons between public and private institutions were supported by two-sided Mann-Whitney U tests and Cliff's δ effect sizes. Nevertheless, the reduced and unbalanced sample sizes, together with the possible selection bias affecting the private-institution sample, may limit statistical power and the generalizability of the findings. Accordingly, statistically non-significant differences should not be interpreted as evidence that the sectors are equivalent. Likewise, statistically significant differences concern only the distributions of the number of subjects mapped to each area and do not demonstrate differences in instructional quality, depth of coverage, student learning, or graduate competence. Accordingly, statistically significant comparisons should be interpreted as evidence concerning the distributions observed in the analyzed samples, whereas non-significant comparisons should be treated as inconclusive rather than as evidence of equivalence. In all cases, the results concern the number of mapped subjects and should not be interpreted as direct evidence of instructional quality or learning outcomes. Moreover, where one sector presents a higher average than the other in a given area, this advantage reflects only the number of subjects mapped to that area. It cannot be concluded from this analysis alone that such a sector provides superior training in these areas, as factors such as course duration, workload distribution, and pedagogical quality are not captured by the curricular mapping. Finally, the corresponding chi-square analyses should be interpreted as exploratory distributional evidence rather than definitive course-level comparisons. Future work should employ course-level proportions or hierarchical models that explicitly account for the nesting of subjects within programs. The conclusions regarding alignment with international curricula are based on a structural comparison of the proportions of knowledge areas. This comparison does not account for differences in pedagogical approach, assessment methods, or the depth of coverage within each area, which may vary considerably across the curricula analyzed. In the analysis of faculty profiles, the sample of public institutions comprises 17 of 28 identified public courses, and the sample of private institutions comprises 16 of 35 mapped private courses. This partial coverage limits the generalizability of the results. Differences in curricular guidelines, evaluation criteria, and training models may lead to distinct patterns of relationship between the teaching profile and the curricular matrix, so the findings should be interpreted with caution.

7 Conclusions

With the increasing complexity of software systems, the demand for qualified professionals in Software Engineering (SE) has intensified. However, training in this area still faces challenges in aligning with industry needs, highlighting curricular issues. In this context, this article analyzed the PPCs and curriculum matrices of 28 SE courses at public institutions and 35 at private institutions across Brazil. A trend toward an increase in the number of courses was identified, with most concentrated in the interior of the states. While public institution courses are distributed across all regions (with a descriptive predominance in the North and Midwest that does not reach statistical significance), private institution courses are significantly concentrated in the South and Southeast ($\chi^2(4) = 23.71$, $p < 0.001$).

Based on the analysis of subjects in relation to the SWEBOK areas, it was evident that in both public and private institutions, there is a concentration of subjects in "Fundamentals

of Computing," "Mathematical Fundamentals," and "Software Construction." In contrast, in both public and private institutions, subjects crucial for professional practice in SE, such as Software Quality, Requirements Engineering, and Software Testing, are less frequently addressed. These findings demonstrate that public and private initiatives share the same curricular gaps in curriculum structure and reinforce the need to review and update curricula to ensure training that is better aligned with market demands and an equitable supply of SE professionals across the country.

According to research, the Brazilian curriculum emphasizes the fundamentals of computer science and programming, a reality evident in both public and private institutions. We also observed that SE courses in Brazil exhibit a more focused structure on SE practices and processes, and a higher frequency of subjects related to management, professional ethics, and writing, indicating greater concern with developing professional and transversal competencies beyond core technical skills, compared with international curricula. However, international curricula feature a strong presence of advanced engineering topics and mathematical fundamentals, which are underrepresented in the Brazilian curriculum.

This study makes three main contributions. First, it provides the first nationwide comparative analysis of Software Engineering undergraduate curricula in Brazilian public and private institutions. Second, it combines the SBC reference curriculum, SWEBOK knowledge areas, and international curriculum references to offer a broader view of curricular structure, emphases, and gaps. Third, it provides empirical evidence that, despite the expansion of Software Engineering programs in Brazil, important areas for professional practice, such as Software Quality, Requirements Engineering, Software Testing, Software Architecture, and Maintenance, remain underrepresented in both sectors.

Based on these findings, we recommend that institutions periodically review their curricula to strengthen these areas, seek a better balance between foundational computing content and core Software Engineering competencies, and adopt stronger alignment with national and international curricular guidelines. We also recommend initiatives that promote greater regional and institutional consistency in Software Engineering education, including curriculum-sharing efforts, accreditation or review mechanisms, and stronger interaction with industry needs.

Future work includes longitudinal analyses to track curriculum evolution over time, qualitative studies with students and faculty to understand the perceived relevance of underrepresented areas, and investigations into the relationship between curricular structure and graduate employability outcomes. Additionally, extending the analysis to other computing-related undergraduate programs, such as Computer Science and Information Systems, would allow for broader comparisons of SE content coverage across Brazilian higher education.

Artifact Availability

The data relating to the PPCs of the courses analyzed, as well as the spreadsheets used, are available in a Zenodo repository: https://doi.org/10.5281/zenodo.15943312

ACKNOWLEDGMENTS

This work was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. Izadora Lima thanks FAPESP (Proc. 2025/04369-6) and FAPEMA (Proc. BIC-18195/26). Rodrigo Santos thanks CNPq (Proc. 316510/2023-8), FAPERJ (Proc. E-26/204.404/2024), and UNIRIO. Awdren Fontão thanks the Brazilian funding agency FUNDECT (Call No. 42/2024). Davi Viana thanks CNPq (Proc. 311533/2025-6 and 448761/2025-4).

References

All India Council for Technical Education (2018). Model curriculum for undergraduate degree courses in engineering & technology. Technical report, All India Council for Technical Education. [Online]. Available: `https://www.aicte-india.org/education/model-syllabus`.

Ardis, M., Budgen, D., Hislop, G. W., Offutt, J., Sebern, M., and Visser, W. (2015). SE 2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering . *Computer*, 48(11):106–109.

Azevedo, T. and da Silva, A. (2023). Ensino de engenharia de dados nas universidades brasileiras: Estado atual e perspectivas de mercado. In *Anais do XXXVIII Simpósio Brasileiro de Bancos de Dados*, pages 432–438, Porto Alegre, RS, Brasil. SBC.

Betz, S. and Penzenstadler, B. (2025). With great power comes great responsibility: The role of software engineers. *ACM Trans. Softw. Eng. Methodol.*, 34(5).

Clementino, A., Lima, E., Lima, L., Guedes, A., Netto, D., and Coutinho, J. (2024). Teaching software engineering: An overview of current approaches and practices in the last decade of sbes. In *Anais do XXXVIII Simpósio Brasileiro de Engenharia de Software*, pages 422–432, Porto Alegre, RS, Brasil. SBC.

da C. Figueiredo, R. M., Jr, L. C. M. R., Ramos, C. S., and Dias, E. (2010). Graduação em engenharia de software versus graduação em engenharia de computação: uma reflexão. In *FEES-Fórum em Educação de Engenharia de Software, Simpósio Brasileiro de Engenharia de Software (SBES)*.

Dierl, S., Howar, F., Mues, M., Naujokat, S., and Schallau, T. (2021). Do away with the frankensteinian programs! a proposal for a genuine se education. In *2021 Third International Workshop on Software Engineering Education for the Next Generation (SEENG)*, pages 26–30.

Ferreira, T., Viana, D., Fernandes, J., and Santos, R. (2018). Identifying emerging topics and difficulties in software engineering education in brazil. In *Proceedings of the XXXII Brazilian Symposium on Software Engineering*, SBES '18, page 230–239, New York, NY, USA. Association for Computing Machinery.

Garousi, V., Giray, G., Tuzun, E., Catal, C., and Felderer, M.

(2020). Closing the gap between software engineering education and industrial needs. *IEEE Software*, 37(2):68–77.

IEEE Computer Society (2024). Swebok guide v4.0.

Mishra, A. and Mishra, D. (2021). *Sustainable Software Engineering: Curriculum Development Based on ACM/IEEE Guidelines*, pages 269–285. Springer International Publishing, Cham.

Oliveira, M., Lima, I., Silva, B., Ayach, F., Menezes, D., dos Santos, R. P., Fontão, A., and Viana, D. (2025). Um relatório de análise dos cursos de graduação em es no brasil: revelações e insights. In *Simpósio Brasileiro de Engenharia de Software (SBES)*, pages 555–565. SBC.

Radaideh, M. A. (2021a). Benchmarking the software engineering undergraduate program curriculum at jordan university of science and technology with the ieee software engineering body of knowledge (software engineering knowledge areas #1–5). In Arabnia, H. R., Deligiannidis, L., Tinetti, F. G., and Tran, Q.-N., editors, *Advances in Software Engineering, Education, and e-Learning*, pages 747–768, Cham. Springer International Publishing.

Radaideh, M. A. (2021b). Benchmarking the software engineering undergraduate program curriculum at jordan university of science and technology with the ieee software engineering body of knowledge: (software engineering knowledge areas 11-15). In *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1043–1049.

Romano, J., Kromrey, J. D., Coraggio, J., Skowronek, J., and Devine, L. (2006). Exploring methods for evaluating group differences on the nsse and other surveys: Are the t-test and cohen'sd indices the most appropriate choices. In *annual meeting of the Southern Association for Institutional Research*, volume 14. Citeseer.

SBC (2017). Currículo de referência da Área de computação, versão 2017. SBC, Acesso em: 29 abr. 2025.

Shaw, M. (2000). Software engineering education: a roadmap. In *Proceedings of the Conference on The Future of Software Engineering*, ICSE '00, page 371–380, New York, NY, USA. Association for Computing Machinery.

Wangenheim, C. G. V. and Silva, D. A. (2009). Qual conhecimento de engenharia de software é importante para um profissional de software. In *Proceedings of the Fórum de Educação em Engenharia de Software*, volume 2, pages 1–8.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., Wesslén, A., et al. (2012). *Experimentation in software engineering*, volume 236. Springer.

Zarour, M., Alenezi, M., and Akour, M. (2025). A framework to evaluate software engineering program using swebok version 4. *Journal of Communications Software and Systems*, 21(1):66–78.