Open Educational Resources for Teaching Computer Science Pedagogy to Secondary School Computer Science Teachers

Torsten Brinda[1][0000−0001−6137−4258], Dieter Engbring[2][0009−0000−0295−2986], Ulrik Schroeder[3][0000−0002−5178−8497], Carsten Schulte[4][0000−0002−3009−4904], Marco Thomas[5][0009−0008−1590−8448], Nadine Bergner[3][0000−0003−3527−3204], and Ira Diethelm[6][0000−0002−5586−8566]

[1] University of Duisburg-Essen, Essen, Germany torsten.brinda@uni-due.de
[2] University of Bonn, Bonn, Germany dieter.engbring@uni-bonn.de
[3] RWTH Aachen, Aachen, Germany
{schroeder,bergner}@informatik.rwth-aachen.de
[4] University of Paderborn, Paderborn, Germany
carsten.schulte@uni-paderborn.de
[5] University of Münster, Münster, Germany marco.thomas@uni-muenster.de
[6] University of Oldenburg, Oldenburg, Germany ira.diethelm@uol.de

Abstract. *FAIBLE.nrw* was a two-year joint initiative of seven German universities that tackled the shortage of secondary-school computer science (CS) teachers by providing openly licensed, research-based learning resources for CS pedagogy in CS teacher education. The consortium created 19 modules for CS teacher education, each pairing a concise theoretical dossier with rich multimedia assets and tested tasks. Constructive alignment guided design, while a ring-review process ensured OER[1] compliance, pedagogical coherence and strong reuse potential. Pilots across partner programs triggered iterative improvements. Released under Creative Commons license on the *https://faible.nrw* portal, the modules span core disciplinary topics and cross-cutting issues, such as programming pedagogy, physical computing, CS teaching methods, and inclusion in CS education, and can be flexibly combined for initial teacher education, in-service up-skilling or certificate courses. By showing how collaborative development, rigorous quality assurance and open licensing accelerate capacity building, *FAIBLE.nrw* offers a transferable blueprint for strengthening CS pedagogy in Germany and beyond.

Keywords: CS teacher education · CS pedagogy · pedagogical content knowledge · secondary CS education · OER

1 Introduction

The accelerating digital transformation of society has led to a widespread advocacy for Computer Science (CS) for all (e. g. *https://www.informaticsforall.org*)

[1] Open Educational Resource

and for an increasing recognition of CS as a foundational component of general education [4]. In Germany, numerous federal states have therefore introduced CS as a compulsory subject in lower secondary school curricula, complementing existing elective offerings in upper secondary education [3]. While this policy shift promises to strengthen pupils' digital competence, it simultaneously exposes a critical bottleneck: the acute shortage of fully qualified CS teachers and CS teacher trainers. Recent projections for the federal state of North Rhine-Westphalia (NRW) in Germany indicate that, without decisive intervention, less than 6% of the anticipated demand for CS teachers will be met by 2030 [5]. Internationally, there are comparable challenges [6, 9]. Addressing this shortfall requires a rapid and sustainable expansion of teacher-education capacity alongside high-quality instructional resources. Within teacher-education programmes, courses in CS pedagogy and CS education research (in German: *Fachdidaktik Informatik*) are pivotal because they bridge disciplinary knowledge and educational science, enabling future teachers to reconstruct complex CS concepts from a pedagogical viewpoint and to adapt them for diverse classroom contexts. In response to current needs, newly established university chairs for CS education research as well as in-service teacher education programmes face the challenge of either rapidly developing high-quality teaching material themselves, a process that is highly time-consuming, or alternatively relying on existing resources. This fragmentation leads to redundant development efforts and missed opportunities for leveraging shared expertise. Open Educational Resources (OER) promise a scalable response. By publishing learning material under permissive licence, educators can translate and adapt content without legal or financial barriers, accelerating dissemination and fostering collaboration. Research shows that modular, research-grounded OER lower the "finding–getting–using" threshold that often prevents instructors from integrating novel topics into their courses [2].

Eight universities in NRW currently provide CS teacher-degree study programmes, with each of them offering at least 15 ECTS[2] credits in CS pedagogy modules through lectures, seminars, and school internships. Five of these universities (together with two universities from other German states) formed the *FAIBLE*[3]*.nrw* consortium to develop OER modules for CS pedagogy in CS teacher education that directly target the aforementioned challenges.[4] The project ran for two years, from 2022 to 2024. The consortium's overarching *aim* was to *create openly licensed, modular resources that link disciplinary CS content with educational science and pedagogy, accommodate heterogeneous programme structures and learner backgrounds, support a spectrum of instructional scenarios from undergraduate to graduate level courses, and to remain adaptable for the use in teacher-training centres as well as in in-service teacher trainings and certificate courses.* Thematic areas such as *programming pedagogy*, *physical com-*

[2] European Credit Transfer System

[3] FAIBLE: Fachdidaktik Informatik in Bausteinen für die Lehre – this paper's title is a translation into English

[4] The consortium was led by the third author. The sixth author was located at the University of Dresden during the project.

puting, *CS teaching methods*, and *inclusion in CS education* were collaboratively selected based on existing local teaching foci, with the aim of ensuring broad curricular relevance rather than institution-specific perspectives. Existing digital assets were systematically refactored, enriched, and relicensed under Creative Commons to accelerate development and promote reusability.

The rest of the paper is structured as follows: section 2 describes related work on OER in CS education. Section 3 outlines the design principles and architecture of the OER modules and section 4 gives an overview of all modules developed during the project. Section 5 reports on first experiences with the modules. Section 6 discusses the broader implications, transferability, and limitations of the initiative.

2 Related Work

Open Educational Resources have gained global attention as a means to improve access to high-quality education. UNESCO's 2019 recommendation on OER emphasises building capacity for stakeholders to create, use, and adapt OER, alongside developing supportive policies and sustainable models, in order to foster inclusive and equitable education [8]. In the context of teacher education, especially in CS, OER hold particular promise for addressing resource gaps and the rapid evolution of digital-skill requirements. Teacher-education programmes worldwide face the challenge of keeping pace with technological change; leveraging openly licensed material is seen as a key strategy to help "bridge the race between education and technology" by enabling continuous, flexible learning for educators. Nationally and internationally, both the research community and practitioners in schools maintain extensive repositories of teaching resources; however, these collections were generally not conceived as OER from the outset. Freely accessible materials that do exist are largely ready-to-use classroom resources. Prominent international portals include *teachcomputing.org*, *exploringcs.org*, and *ai4k12.org*, as well as *csunplugged.org* and *barefootcomputing.org* for younger learners. In Germany, a variety of material can be found on platforms such as *informatik.schule.de* and several states' school servers (e. g., Hamburg: *https://bildungsserver.hamburg.de/informatik/unterricht/unterrichtsmaterial*). In addition, several university groups have set up extensive online collections of their own materials (e. g., University of Wuppertal: *https://ddi.uni-wuppertal.de/archiv/madin//material/materialsammlung/index.html*, FU Berlin: *computing-education.de*, University of Göttingen: *ddi-mod.uni-goettingen.de*, RWTH Aachen: *schuelerlabor.informatik.rwth-aachen.de*). While these examples are hands-on resources and highly valuable for a science-based education in CS pedagogy, they cannot replace a systematic academic foundation. Freely accessible, research-informed modules on CS pedagogy are available only sporadically. One notable example is Peer Stechert's YouTube channel (*https://www.youtube.com/channel/UCw4vXnOoHHNSwJGfgOVf2DQ*), which summarises various questions and topics in CS pedagogy in concise video format. Several studies have examined the use and needs of OER in CS education. An early survey of CS instructors

by Dichev and Dicheva found strong interest in OER but also significant hurdles in practice [2]. A primary difficulty reported was the "finding–getting–using" challenge: with the proliferation of OER repositories, educators struggle to locate relevant material when needed. Additional barriers noted include the effort required to adapt materials to specific teaching contexts and a lack of appropriate content in certain topic areas. Notably, language was identified as a barrier for non-English-speaking instructors, since many OER were available only in English. These findings highlight the importance of improving searchability, providing localisation, and expanding the pool of high-quality OER content in various languages. Since then, the OER movement in CS and STEM education has grown, including dedicated initiatives in the German-speaking (DACH) region. For example, after the discontinuation of the long-running German CS education journal *LOG IN* in 2022, the community launched a new open-access journal *Informatische Bildung in Schulen (IBiS) (in English, Computer science education in schools)* to continue sharing CS education practice and research for teachers (see *https://www.informatischebildung.de*). All IBiS articles are published under open Creative Commons licenses, reflecting a commitment to OER principles. This enables CS teachers to access up-to-date pedagogical insights and classroom material freely, and to adapt them for their own teaching. Another German project, the *MINT-L-OER-amt* initiative at RWTH Aachen, has explored OER integration into STEM teacher training, yielding lessons about sustaining OER use beyond initial funding. All these efforts align with broader calls in Germany to modernise teacher preparation through digital material and open content [1]. Recent research underscores how educators are taking advantage of OER in practice. Rampelt et al. (2025) conducted a cross-sector survey of 260 educators in Germany, Austria, and Switzerland who use OER from an AI-education platform [7]. They found that teachers highly value "smaller, modular OER formats" that can be flexibly integrated into their teaching and prioritise resources that are high-quality, accessible, and curriculum-relevant. Interestingly, the reputation of the source was less important than the fitness of the content for teachers' needs. Across sectors, modularity and practical applicability were key factors for adoption, reinforcing earlier observations that ease of finding and adapting resources is critical. In summary, prior work and initiatives, together with the diverse but often non-OER repositories noted above, demonstrate both the potential and the unmet needs surrounding OER for CS teacher education. Given the slow incorporation of digital and CS topics in formal teacher-training curricula [1], coordinated efforts (such as the project described in this paper) are helpful to realise the full benefits of OER in CS teacher education.

3 Design of the CS Pedagogy OER Modules

The complex challenges outlined in section 1, from the urgent need for CS teacher-education capacity to the pronounced heterogeneity of existing programmes, require a systematic response that goes beyond isolated resource creation. Informed by these contextual demands and the practical and theoretical

groundwork described in section 2, the consortium developed an OER ecosystem designed to address structural, curricular, and pedagogical constraints across institutions. The goal was not to produce isolated materials but to establish a coherent and adaptable framework for supporting CS pedagogy in initial and continuing teacher education. The project's starting point was the recognition that the participating universities had already accumulated a broad range of teaching and learning materials in CS pedagogy and in CS education research. However, these materials were typically tailored to local curricular structures, embedded in institution-specific learning management systems, and not openly licensed. Moreover, they were originally created for in-person instruction and not developed with interoperability or modular reuse in mind. Given this situation, the project set out to select a cross-institutional set of topic modules that were pedagogically relevant across sites and amenable to conceptual refinement. A particular emphasis was placed on aligning these modules with didactic needs in CS education, preparing them for open publication under Creative Commons licenses, and ensuring that they could be meaningfully reused across different programme types and institutional settings. To ensure curricular breadth and foster shared ownership, the thematic focus of the modules was collectively determined by the consortium. Topics such as *Programming Pedagogy*, *Physical Computing*, *CS Teaching Methods*, and *Heterogenity and Inclusion in CS Education* were deliberately chosen not only for their relevance to teacher education but also for their ability to transcend local institutional priorities. Existing digital artefacts, e.g., slides, activities, and conceptual guides, were systematically reviewed, adapted, and enriched where possible, with licensing barriers removed to ensure compliance with OER standards. These efforts aimed to combine the practical strength of existing teaching resources with a common quality framework grounded in academic and didactic research. In addition to the content dimension, the project had to address structural and operational considerations. Since participating universities differed substantially in their degree programmes, contact-hour models, and school internships integration, the materials needed to be usable in both low- and high-contact scenarios, across undergraduate and graduate levels, and in both university-based and school-centred contexts. This variability necessitated a modular design approach, flexible sequencing, and metadata structures that allow filtering according to programme type, didactic function, and prior knowledge requirements. Furthermore, long-term sustainability and transferability were to be embedded into the design through reciprocal usage commitments between institutions and through structures that facilitate future adaptation and community contribution. By combining these content-related, didactic, and operational dimensions, the project established the foundation for a research-informed, openly licensed resource infrastructure.

In the following, the resulting design principles and requirements that guided the development of individual modules and the overall ecosystem architecture are described in detail. These principles address the barriers identified in the literature, such as limited discoverability, lack of localisation, and insufficient coupling of theory and practice, and provide a coherent basis for sustained and

scalable implementation of OER in CS teacher education. To ensure didactic coherence and foster meaningful integration of theory and practice, the design of each module had to be *guided by the principle of constructive alignment*. Modules need to address at least one central didactic dimension, such as learning objectives, learner prerequisites, content selection, or instructional methods, and illustrate how choices in one dimension influence others. Learning outcomes are expected to go beyond knowledge recall and support the ability to classify theoretical constructs, apply them to authentic classroom scenarios, analyse interdependencies among didactic decisions, derive practice-oriented implications from theory, and formulate criteria for lesson evaluation. Assessment formats should be aligned accordingly and include portfolios, written assignments, or oral examinations that provide evidence of the ability to transfer educational theory into classroom practice. Next, the material had to be designed *in accordance with didactic-methodological principles that support self-organised and self-directed learning*. The modules should function both as stand-alone self-study units and as components within face-to-face teaching formats, particularly those that allocate in-person time for reflection and application, such as inverted classroom models. The instructional design should facilitate the embedding of modules in various formats including lectures, exercises, seminars, and laboratory work. It should also give instructors the flexibility to combine several modules into one course and to adapt or extend them with site-specific content in line with local curricular or institutional priorities. In line with international recommendations for inclusive digital learning environments, the project should provide a *diverse portfolio of media formats*, including text-based resources, slidecasts, videos, interactive H5P activities (see https://h5p.com), diagrams, interviews, quizzes, and automated self-assessments. All materials had to comply with established e-learning design standards and to make use of Universal Design for Learning (UDL) guidelines to ensure accessibility and ease of use. The inclusion of learner-generated content should be encouraged to allow for immediate user feedback and to model collaborative practices in open educational resource development.

The intended implementation of the modules across partner institutions required a *coordinated workflow and a shared framework* to ensure consistency while supporting local adaptability. Each module had to be designed to cover one to three teaching sessions and allow for flexible sequencing into longer instructional units. A joint instructors' handbook should provide guidance on inter-module dependencies and recommend effective sequencing strategies. Coordination mechanisms, such as regular consortium meetings, are needed to manage variant versions where curricular heterogeneity demands differentiation. Moreover, the design should facilitate cross-institutional collaboration, including guest contributions from partner universities and opportunities for inter-site exchange among students working with shared modules. By integrating constructive alignment, learner autonomy, and media diversity, the OER ecosystem directly addresses the main barriers highlighted in the previous sections – namely, scarce research-based resources, limited localisation options, and insufficient theory–practice integration. Its modular architecture and open licensing en-

able rapid adaptation for continuing-education programmes, certificate courses, and teacher-training centres, thereby supporting the scalable expansion of CS teacher-education capacity envisioned in section 1.

4 Overview of the CS Pedagogy OER Modules

As a result of the *FAIBLE.nrw* project, a curated set of 19 modules was developed. All modules are published in German language on the project website *https://faible.nrw* and on the portal *ORCA.nrw*, see *https://www.orca.nrw/oer/oer-finden/gefoerderte-kurse/oercontent-nrw/faible/*. Moreover, an open Moodle course is provided, which contains all modules, see *https://moodle.rwth-aachen.de/course/view.php?id=26921*.

The developed modules (see Table 1) are intended for use in on-site university courses, in teacher training programmes, and as supplementary material for self-study. In academic teaching contexts, the material can be employed either selectively or in its entirety to address typical focal areas of CS education. Most of the modules are designed to cover a standard 90-minute teaching unit, although partial use to complement existing instructional material is also feasible. Some of the modules may also extend across multiple teaching sessions. Particularly extensive are the modules on *Programming Pedagogy* and *Inclusion in Practice*, which provide materials and concepts suitable for longer instructional sequences. As a result of the coordination process among the project partners during the module development, five thematic focal areas evolved. The area *Foundational Perspectives on CS Education* includes modules that elucidate the nature and essential dimensions of CS and present various subject-specific pedagogical approaches that are typically discussed in German-speaking CS pedagogy courses. The area *Teaching Practice in CS Education* comprises modules that adopt a CS-specific perspective on questions of concrete lesson planning. Important topics of school-level CS can be found in the area *Subject-Specific Content in CS Education*. The material on *Programming Pedagogy* is particularly extensive, reflecting the central importance of the associated questions in subject-specific educational practice. The fourth area encompasses modules addressing *Heterogeneity, Inclusion, and Educational Equity in CS Education*. The need to address such issues in teacher education has been formally anchored in federal teacher training regulations. The underrepresentation of women in CS is another major issue for which teachers must be sensitised. In NRW in Germany, many regions are characterised by migration-related heterogeneity, which creates additional demands for effective approaches to language development. The final area, *Open Educational Resources and Infrastructure*, provides guidance for the creation of OER material and is intended as a support tool for future OER developments in this field.

A detailed overview of all modules (with information about each module's content, the logical line of thought, provided material, intended use cases, time requirements and other aspects) can be found on the website *https://faible.nrw/bausteine/*.

Table 1: Overview of the CS Pedagogy Modules

Module	**Content**	**Included Materials**	**Time**
Foundational Perspectives on CS Education			
What is Computer Science?	Historical genesis of CS from hardware, software, and theoretical roots	Slides, screencasts, exercises with solutions	45-90 mins.
Historical Models of CS Education	Classical models of CS teaching (hardware-, algorithm-, application-, user-oriented)	Slides, screencasts, video, exercises	45-90 mins.
Idea Orientation	Fundamental ideas of CS; selection criteria (horizontal, vertical, sense, time, and goal criterion)	Slides, screencast, embedded video, interactive H5P quiz	45-90 mins.
Model Orientation	Didactic approach of model orientation in CS; connects general model theory (Apostel, Stachowiak) with modelling in CS; relates modelling concepts to upper secondary CS curricula	Slides, screencasts, video, exercises	45-90 mins.
System Orientation	Didactic approach of system orientation; analysis, development and evaluation of computing systems as a focus of CS teaching	Slides, website with expert interview (video, audio, transcript)	90-180 mins.
Digital Education	Concept and increasing importance of digital education, relates digital education to policy frameworks	Slides with audio commentary, transcript, exercises, quiz, additional reading material	60-90 mins.
Teaching Practice in CS Education			
Context-Oriented CS Education	Didactic approach of context-oriented CS teaching; compares general context-oriented approaches (incl. science education); introduces the "Informatics in Context (IniC)" framework	Slides	45-90 mins.
CS Teaching Methods	Collection of teaching methods for different classroom scenarios; differentiates guiding methods and phase-specific methods	Method cards, overview document, explanatory script	45-90 mins.
Educational Reconstruction	Process of educational reconstruction for transforming content for teaching	Slides, screencasts, audio file, transcript, exercises	45-90 mins.
CS-Pedagogical Principles	Relates CS-pedagogical principles to general features of good teaching (e.g. Meyer); encourages principled reasoning in lesson planning	Slides, screencast	45-90 mins.

Table 1: Overview of the CS Pedagogy Modules (continued)

Module	**Content**	**Included Materials**	**Time**
Practical Tips	Typical practical challenges from everyday school life (e.g. blackboard work, exams), provides experience-based solutions and reflections	Slides, screencasts	90-180 mins.
Subject-Specific Content in CS Education			
Programming Pedagogy	Role of programming in CS education; theories, approaches, and tools for introducing programming; pair programming	Slides, practical programming exercises (GUIs, Greenfoot, data structures, sorting)	multiple units
Physical Computing	Physical computing with a focus on micro controllers (e. g. Arduino); revisits basic electronics and technical CS concepts relevant for school, smart-home project example	Slides, screencasts, transcript, exercises	90-180 mins.
Data Awareness	Data awareness for data-driven digital artefacts; explanatory model of the role of data in such artefacts	Slides, integrated tasks and discussion prompts	90-180 mins.
Heterogeneity, Inclusion, and Educational Equity in CS Education			
Inclusion with a Focus on Visual Impairments	Foundations of heterogeneity and inclusion in schools; assistive technologies, differentiation, and pedagogical frameworks (e.g. UDL)	Slides, integrated exercises	90-270 mins.
Inclusion in Practice	Hands-on concept for a course on inclusion in CS education; phases: studying inclusion basics, creating support materials, designing inclusive prototypes	Moodle learning space describing concept and phases	multiple units
Gender	Gender-related aspects in CS (education); aims at raising awareness and professional responsibility of (future) teachers	Slides with audio commentary, transcript	90-180 mins.
Discipline-Specific Language Education	Language education in CS with a focus on writing; policy demands, theoretical foundations, and language-related challenges; concrete support strategies	Slides, integrated exercises	90-180 mins.
Open Educational Resources and Infrastructure			
OER Cycle	Self-study course introducing the concept and practice of Open Educational Resources (OER)	Stand-alone Moodle course or course section on OER, H5P quiz	not specif.

5 First Experiences

A first evaluation of the OER modules followed a jointly developed concept, which combined formative and summative components. This included an internal ring peer review across consortium sites, integration of modules into university teaching with accompanying course evaluations, and expert inspections of module usability. The evaluation focused on three core criteria: compliance with OER standards, pedagogical quality, and reusability across institutional contexts.

Each module underwent peer review by at least two experts from partner institutions. Reviewers used a standardised rubric to assess compliance with OER licensing requirements, content quality, modularity, and usability in heterogeneous teaching environments. Overall, the review process confirmed that all modules met the core requirements for open licensing and academic integrity. Reviewers positively acknowledged the didactic structure, clarity of learning objectives, and integration of diverse media formats in most modules. Nevertheless, several recurring challenges were identified. These included incomplete or inconsistent attribution of third-party media (e.g., images), minor inconsistencies in layout or formatting across slides, insufficient metadata or orientation aids for new instructors, and occasional misalignment between intended learning outcomes and provided activities. In a few cases, technical aspects, such as video narration quality or the lack of time estimates for exercises, were flagged as areas for improvement. All feedback was compiled and addressed in two coordinated revision cycles that concluded in the second quarter of 2024. Revisions included harmonisation of design templates, correction of licensing attributions, addition of didactic annotations and example applications, refinement of self-assessment elements, and the development of brief module overviews to support onboarding by new users.

From winter semester 2022/23 to 2024, the modules were piloted across the five participating universities in NRW and integrated into diverse teaching formats, including lectures, seminars, and practicum sessions. Standard course evaluations administered at the local level provided feedback from student teachers, which guided further refinements. Commonly requested improvements included shorter video segments, additional self-check activities, and clearer contextualisation of content within school curricula. While no consolidated quantitative data were available, qualitative feedback was consistently positive regarding the accessibility, relevance, and adaptability of the material.

Given the decentralised architecture of the OER ecosystem and the absence of a dedicated learning management platform, formal usability testing was substituted by expert walkthroughs and inspections. These were conducted before initial deployment and focused on navigation, media coherence, and clarity of instructions. The findings, documented in the review materials, informed subsequent design improvements and interface enhancements.

In sum, the evaluation confirmed that the developed modules meet high standards of didactic design, reusability, and open licensing. The evaluation process

also enabled iterative improvements and laid the foundation for evidence-based integration of OER into the broader teacher education infrastructure.

6 Discussion

The *FAIBLE.nrw* project demonstrates that collaboratively developed, openly licensed modules can address the long-standing shortage of research-grounded resources in computer-science teacher education. By publishing a complete portfolio on *https://faible.nrw* and by aligning theoretical dossiers with concrete classroom tasks, the project responds directly to the "finding–getting–using" challenge highlighted in section 2. The ring evaluation confirms that the materials are OER-compliant, reusable across heterogeneous curricula, and sufficiently detailed for immediate classroom adoption.

The modular architecture of the material enables selective integration in multiple contexts:

- *CS teacher degree study programmes at universities:* Universities can embed single modules (e. g. *Programming Pedagogy*, *Physical Computing*) into existing lectures or assemble a coherent sequence for an entire course on CS pedagogy. Modules such as *CS Teaching Methods* provide ready-made artefacts for reflection seminars that accompany school internships.
- *Continuing professional development:* Teacher-training centres may offer blended workshops that pair self-study with on-site coaching, using the application tasks and H5P quizzes for formative assessment.
- *Certificate and extension programmes:* In-service teachers acquiring an additional CS qualification can work through the portfolio as a structured self-learning path.
- *Cross-university collaboration:* Because each module includes editable source files, instructors from different institutions can adapt slides, videos, and quizzes to local curricula and share improvements back to the portal.

All modules were produced in German language to meet the immediate needs of teacher-education programmes in NRW. Nonetheless, the open licensing explicitly permits translation. Emerging AI-based translation tools, such as *DeepL*, or general-purpose AI chatbots can automate a first pass, after which domain experts could perform quality assurance. Segment-level translation of slide decks and H5P activities is facilitated by the modular file structure; metadata keys (title, learning objectives, licence) require only minimal manual editing. Such a workflow paves the way for transferring the materials to the wider international CS education community.

The project's evaluation relied on expert review and standard course feedback; no large-scale, cross-site learning-outcome study was conducted. Future research could employ quasi-experimental designs to measure gains in pedagogical content knowledge across multiple cohorts. Moreover, while module uptake has begun at two external universities, systematic monitoring of broader adoption – and of community-driven revisions – remains an open task.

The *FAIBLE.nrw* project illustrates a viable pathway for rapidly expanding teacher-education capacity: leverage existing academic expertise, apply constructive-alignment principles, and disseminate outputs under open licences. By coupling these elements with translation-friendly formats, the project not only serves the German context but also lays groundwork for international collaboration in CS teacher education.

Acknowledgements

FAIBLE.nrw was supported by the Ministry of Culture and Science in NRW in the funding line *OERContent.nrw* (reference number 214-76.01.08-25-1820).

References

1. Brinkmann, B.: Comparative data shows little progress in the acquisition of digital skills on teacher training programmes. CHE (Centre for Higher Education) Press Release (2021), https://www.che.de/en/2021/vergleichsdaten-zeigen-kaum-fortschritte-beim-erwerb-digitaler-kompetenzen-im-lehramtsstudium/
2. Dichev, C., Dicheva, D.: Open educational resources in computer science teaching. In: Proceedings of the 43rd ACM Technical Symposium on Computer Science Education. p. 619–624. SIGCSE '12, ACM, New York, USA (2012). https://doi.org/10.1145/2157136.2157314
3. Gesellschaft für Informatik e. V., Stifterverband, Heinz Nixdorf Stiftung: #InformatikMonitor – Zur Situation des Informatikunterrichts in Deutschland (October 2024), https://informatik-monitor.de/2024-25
4. Hubwieser, P., Giannakos, M.N., Berges, M., Brinda, T., Diethelm, I., Magenheim, J., Pal, Y., Jackova, J., Jasute, E.: A global snapshot of computer science education in k-12 schools. In: Proceedings of the 2015 ITiCSE on working group reports, pp. 65–83 (2015)
5. Klemm, K.: Lehrkräftemangel in den MINT-Fächern: Kein Ende in Sicht. Zur Bedarfs- und Angebotsentwicklung in den allgemeinbildenden Schulen der Sekundarstufen I und II am Beispiel Nordrhein-Westfalens (2020), https://www.telekom-stiftung.de/sites/default/files/mint-lehrkraeftebedarf-2020-ergebnisbericht.pdf
6. Ramagoni, S.G.: Understanding computer science teacher capacity. In: Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 2. p. 623–624. ITiCSE 2023, ACM, New York, USA (2023). https://doi.org/10.1145/3587103.3594150
7. Rampelt, F., Ruppert, R., Schleiss, J., Mah, D., Bata, K., Egloffstein, M.: How do AI educators use open educational resources? a cross-sectoral case study on OER for AI education. Open Praxis **17**(1), 46–63 (2025). https://doi.org/10.55982/openpraxis.17.1.766
8. UNESCO: Recommendation on open educational resources (oer). UNESCO General Conference Recommendation (2019), https://unesdoc.unesco.org/ark:/48223/pf0000370936
9. Williams, H.E., Williams, S., Kendall, K.: Cs in schools: Developing a sustainable coding programme in australian schools. In: Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education. p. 321–327. ITiCSE '20, ACM, New York, USA (2020). https://doi.org/10.1145/3341525.3387422