Bridging the Gap: A Development Approach for Seamless Educator-Developer Collaboration in E-Learning

Master's Thesis of

Daniel Augustin

At the KIT Department of Informatics
KASTEL – Institute of Information Security and Dependability

First examiner:	Prof. Dr.-Ing. Anne Koziolek
Second examiner:	Prof. Dr. Ralf H. Reussner
First advisor:	M.Ed. Kai Marquardt
Second advisor:	Dr.-Ing. Lucia Happe

01. Feburary 2024 – 01. August 2024

I declare that I have developed and written the enclosed thesis completely by myself. I have not used any other than the aids that I have mentioned. I have marked all parts of the thesis that I have included from referenced literature, either in their original wording or paraphrasing their contents. I have followed the by-laws to implement scientific integrity at KIT.

Karlsruhe, 01.08.2024

Omitted for publication

. . . .

(Daniel Augustin)

Abstract

E-learning is pedagogy empowered by technology. Many development approaches emphasize collaboration between educators and developers during e-learning course development. This thesis builds on these approaches by highlighting this collaboration. A collaborative e-learning course creation platform was envisioned and partially implemented, with an evaluation identifying strengths and weaknesses through interviews and surveys with educators. Results indicate educators appreciate the concept but are hesitant to use the platform due to its early development stage. Areas for improvement and future features were identified, offering valuable feedback for future development and emphasizing the need to combine pedagogical and technological expertise. These findings provide valuable feedback for future development efforts to create a functional and user-friendly e-learning course creation platform.

Zusammenfassung

E-Learning ist durch Technologie unterstützte Pädagogik. Viele Entwicklungsansätze betonen die Zusammenarbeit zwischen Pädagogen und Entwicklern bei der Erstellung von E-Learning-Kursen. Diese Arbeit baut auf diesen Ansätzen auf, indem sie die Zusammenarbeit vertieft untersucht. Eine kollaborative Plattform zur Erstellung von E-Learning-Kursen wurde konzipiert und teilweise implementiert. Durch Interviews und Umfragen mit Pädagogen wurden Stärken und Schwächen der Plattform aufgezeigt. Die Ergebnisse zeigen, dass Pädagogen das Konzept schätzen, aber aufgrund des frühen Entwicklungsstadiums zögern, die Plattform zu nutzen. Identifizierte Verbesserungsbereiche und zukünftige Funktionen bieten wertvolles Feedback für die Weiterentwicklung und betonen die Notwendigkeit, pädagogische und technologische Expertise zu kombinieren.

Contents

List of Figures

List of Tables

1. Introduction

The rapid evolution of technology in recent years has brought about significant changes in the field of education. E-learning, as an innovative method for delivering education independent of time and location has gained prominence [19]. However, despite the promising opportunities that e-learning can offer, it faces challenges that profoundly impact its successful implementation [6]. So far, e-learning has not experienced a breakthrough success. Initial expectations that e-learning would revolutionize teaching and learning processes have encountered obstacles.

There are many definitions of e-learning [34]. One misconception is viewing e-learning simply as the transfer of existing teaching materials to the internet [6]. The superficial application of cutting-edge technology alone is insufficient. Rather, a solid foundation in pedagogical considerations adaptable to the evolving educational landscape is required [26]. Therefore, e-Learning should be defined as pedagogy empowered by digital technology [41].

Historically, educators have controlled the pace, place, time and style of presentation and interaction [6]. Well-designed e-learning courses extend these boundaries and allow learners to become actively involved in their learning processes. However, a poorly designed e-learning courses can be just as rigid and dogmatic and non-interactive as poorly taught face-to-face courses [26]. Thus, e-learning requires a paradigm shift not only for learners, but also for educators, administrators, and technical staff [6, 26]. The paradigm shift extends not only to the pedagogical level but also to software development. Traditional software development approaches prove inadequate as they do not address the specific teaching and learning issues of e-learning [19]. Educators may be resistent to change, especially when the focus is on technological capabilities rather than the learning process [58]. Effective collaboration between educators and developers is essential to design effective e-learning courses. The synergy between pedagogical expertise and technological proficiency requires seamless integration of both domains [19]. Educators bring valuable insights into instructional design, learning theories, and learner needs. Developers contribute technical expertise to translate these insights into functional and user-friendly e-learning platforms [26, 41]. The success of e-learning depends on educators being able to focus on their pedagogical role without the burden of dealing with technical details [35].

Despite challenges and uncertainties, there are indications that e-learning will continue to expand. The merging of pedagogical expertise and technological know-how could soon make e-learning an integral part of education. The 'e' in e-learning might eventually disappear as technology seamlessly integrates into education [41]. The goal should be to strengthen collaboration between educators and developers to further enhance the effectiveness of e-learning.

However, collaboration in e-learning course development presents significant challenges. The interdisciplinary nature of e-learning requires the coordination of diverse teams, including educators, developers, instructional designers, and domain experts [12]. Each group brings its own perspectives, goals, and terminologies, which can lead to communication barriers and misunderstandings. Educators may lack the technical knowledge to effectively convey their pedagogical requirements, while developers might struggle to understand the educational context and learner needs. Moreover, aligning the schedules and priorities of these multidisciplinary teams can be difficult. Educators and developers often operate under different time constraints and work cultures [20]. These challenges highlight the need for clear communication channels, well-defined roles, and a shared vision among all stakeholders to successfully develop and implement effective e-learning courses.

To achieve this goal, this thesis aims at deriving an e-learning course development approach that focuses on seamless educator and developer collaboration. A supporting application is developed to further lower the barrier of entry. This is especially useful for non-technical stakeholders like educators. The stakeholders should be able to collaborate without having to dive too deep into the specifics of the others domain. This thesis aims to answer the following research questions.

1. What are the requirements for a collaborative e-learning development approach to ensure the alignment of pedagogical goals and software development processes?
2. In what ways does the proposed development process fulfill the individual requirements of the main stakeholders educators and developers?
    a) What are the key features and functionalities that educators and developers, find most valuable and essential in an application designed to support collaborative e-learning development?

Two main tasks can be identified for this thesis. Deriving a collaborative development approach is the foundation for the later developed application. Since many development approaches have been proposed so far, this thesis adapts existing approaches and extends them so the collaboration aspects are highlighted. The second task is concerned with developing the application. The application is developed simultaniously to the first task. An iterative process is used to react to changing requirements from the first task. The development approach and application are evaluated in a case study. The case study is uses surveys and interviews to gauge satisfaction with functionality, applicability, and usability.

Chapter 2 introduces the necessary foundations for the following chapters. Already existing e-learning course development approaches are presented in chapter 3. The insights from the related work are used to derive the collaborative development approach in Chapter 4. The vision and implementation of the e-learning course creation application are described in Chapter 5. The evaluation of the proposed development approach and application is performed in Chapter 6, and the results are discussed in Chapter 7. Lastly, Chapter 8 concludes the results and outlines future work.

2. Foundation

This chapter describes foundational concepts essential for understanding the broader themes and methodologies discussed in the following chapters.

2.1. Learning Theories

Learning is a multifaceted and dynamic process at the core of education. It involves the retention and transfer of knowledge to new and different situations [34]. Learning can be described through various dimensions, each influencing the design and implementation of instructional strategies. The temporal dimension involves the timing aspects of learning, including synchronization (simultaneous activity of learners and educators) and the available learning time. Learning can be synchronous (e.g., live seminars) or asynchronous (e.g., distance learning), with flexibility in the time available for learning activities. The spatial dimension addresses the physical location of learners and educators. Learning can occur locally (in the same place) or in a distributed manner (different locations), with technology bridging the gap to facilitate remote interactions. The information delivery dimension differentiates between person-centered and media-centered learning. Person-centered learning emphasizes the role of educators or tutors, while media-centered learning focuses on the use of learning media to deliver content [49]. Learning theories provide conceptual frameworks to understand and interpret how individuals acquire knowledge, skills and competencies [19]. Effective e-learning requires careful consideration of the underlying pedagogy. Theories such as behaviorism, constructivism, and collaboration provide essential frameworks for designing instructional materials and activities [51]. This chapter introduces these learning theories and their implications for instructional design in e-learning.

The behaviorist learning theory, rooted in the work of Skinner, emphasizes the efficient transmission of knowledge from educators to learners. It views learning as a process of conditioning, where reinforcement of specific behaviors increases the likelihood of their recurrence. In behaviorist settings, educators play a central role, fostering stability and certainty in knowledge acquisition. This approach is beneficial for novice learners requiring foundational knowledge but is critiqued for limiting opportunities for deeper cognitive engagement and learner expression [29, 19].

In contrast, constructivist learning theory states that knowledge is actively constructed by learners through their experiences and interactions. Learning is viewed as an active process where learners build new knowledge upon the foundation of prior knowledge. This learner-centered approach positions educators as facilitators who guide learners in practical problem-solving activities. Constructivist learning encourages critical thinking and the application of knowledge in varied contexts [19, 21]. It promotes interactive learning, where learners actively participate and influence the course of their learning. Constructivism has emerged as a dominant approach to learning [51].

The collaborative learning theory emphasizes the role of social interaction in the learning process. It involves dynamic interactions among learners and educators which fosters discussion, dialogue, and information sharing. This theory highlights the importance of relationships and collaboration in knowledge construction [19]. This makes it particularly relevant in e-learning environments where digital technologies can facilitate group learning activities.

Learning theories provide valuable insights into how knowledge is acquired and applied. By understanding these theories, educators can design effective e-learning environments that cater to diverse learner needs and promote meaningful learning experiences. Instructional design emerged as a scientific and technological sub-discipline of educational psychology. The core idea of instructional design is the acknowledgment that there is no single, universally correct teaching method. Instead, the appropriate learning environment is defined based on different categories of learning tasks and various learning prerequisites. This perspective recognizes that learning theories and their associated methods are not inherently unsuitable but should not be used as standalone methods universally. Instructional design emphasizes creating tailored learning environments that address specific learning objectives and learner needs [49, 51]. As digital technologies continue to evolve, the integration of sound pedagogical principles remains crucial in enhancing the quality and effectiveness of e-learning [33, 21]. By leveraging a combination of instructional approaches, educators can develop strategies that effectively promote different types of learning performances, ensuring that e-learning environments are both engaging and effective.

2.2. E-Learning

The rapid evolution of technology in recent years has brought about significant changes in the field of education. E-learning, as an innovative method for delivering education independent of time and location has gained prominence [19]. However, despite the promising opportunities that e-learning can offer, it faces challenges that profoundly impact its successful implementation [6]. Initial expectations that e-learning would revolutionize teaching and learning processes have encountered obstacles.

Numerous attempts have been made to define e-learning, each emphasizing different aspects [34]. One common definition highlights its freedom from temporal and spatial constraints, allowing learners to access educational content from anywhere, at any time. Other definitions focus on the technical infrastructure, emphasizing the role of networking and the internet in delivering e-learning materials. This is in line with definitions that view e-learning as synonymous with terms like web-based learning (WBL), internet-based training (IBT), advanced distributed learning (ADL), web-based instruction (WBI), online learning (OL), and open/flexible learning (OFL) [34]. This synonymity illustrates the variety of terminology, all of which are based on the use of network technologies. E-learning is also characterized as a learner-centered approach that prioritizes interactivity, self-pacing, repetition, and customization. This design aims to cater to the individual needs and preferences of learners, showcasing the flexibility inherent in e-learning environments. Furthermore, e-learning is viewed as a new educational paradigm aimed at fostering a culture of knowledge sharing within organizations. This perspective emphasizes its role in transforming organizational learning and cultivating a knowledge-driven culture [34].

In summary, e-learning encompasses several defining characteristics:

1. Internet is used as its primary platform
2. Information is distributed through networked courses
3. Global distribution and sharing of educational resources is possible
4. Method of studying
5. Flexibility in terms of study time and location

These characteristics are helpful to understand the technological aspects and resulting opportunities. However, technology is only one critical success factors of effective e-learning [34]. A misconception is viewing e-learning simply as the transfer of existing teaching materials to the internet [6, 51]. The superficial application of cutting-edge technology alone is insufficient. Rather, a solid foundation in pedagogical considerations adaptable to the evolving educational landscape is required [26, 43]. Educators have traditionally controlled many aspects of the learning environment, such as pace, place, time, and style of presentation and interaction [6]. However, well-designed e-learning courses can expand these boundaries, encouraging active learner participation. Conversely, poorly designed e-learning courses can be as inflexible and non-interactive as ineffective face-to-face courses [26, 41]. Traditional software development methods often fail to address the unique teaching and learning issues inherent in e-learning [19]. Resistance to change among educators can be a significant barrier, especially when the emphasis is on technological capabilities rather than the learning process [58]. Educators should focus on instructional design, learning theories, and understanding learner needs, while developers handle the technical aspects, translating these insights into practical solutions [26, 41].

The dependence between pedagogy and technology is illustrated in Table 2.1. E-learning, derived from the combination of electronic and learning, is fundamentally guided by pedagogical principles. Effective e-learning requires a well-defined pedagogical framework to direct the use of technology. Without a solid pedagogical foundation, technological tools can become ineffective. Conversely, if the technology employed is unreliable or overly complex, the e-learning experience can become frustrating for both educators and learners. The tendency to assume that superior technology will inherently lead to better education, known as technocentrism [45], is a common misconception among some technology enthusiasts [41]. However, the

	Sound pedagogy	Unsound pedagogy
Reliable, easy-to-use technology	Effective e-learning	Technocentrism
Unreliable, complex-to-use technology	Frustration	Disaster

Table 2.1.: E-learning as a dependent function [41].

essence of good teaching remains unchanged, regardless of the medium through which it is delivered.

E-learning should be seen as a versatile educational tool that can be integrated within various education models, such as on-campus and distance learning. E-learning supports different learning theories like behaviorism and constructivism. It serves as a means rather than a mode of education, utilizing technology to enhance learning across different contexts [41]. Making the most of technology involves leveraging these features, rather than simply using digital means to replicate traditional methods. Without this, technology risks being used merely to enhance conventional learning designs, rather than creating more effective and innovative ones [33]. By prioritizing pedagogical goal over the choice of specific technologies, e-learning has the chance to enable innovative forms of education that blend the strengths of traditional and distance learning. E-learning advances through pedagogical innovation and requires careful consideration of how learners will interact with the provided opportunities. The core educational process remains unchanged, and the success of e-learning depends on its ability to address identified educational needs [49, 41, 51]. Thus, the definition of "e-learning is pedagogy empowered by digital technology" [41, p. 2] will be used as a guide throughout this thesis.

2.3. Blended Learning

In 2003, the American Society for Training and Development identified blended learning as one of the top ten trends to emerge in the knowledge delivery industry. Blended learning, also referred to as hybrid or mixed learning, combines online and face-to-face (F2F) instruction, integrating the strengths of both traditional and distributed learning systems. This approach emphasizes the central role of computer-based technologies while maintaining the pedagogical richness. However, it is crucial to recognize that without proper design, blended learning can inadvertently combine the least effective elements of both models, potentially diminishing its effectiveness [17].

Blended learning is chosen overwhelmingly for three primary reasons: improved pedagogy, increased access and flexibility, and enhanced cost effectiveness [17]. It challenges the status quo by maintaining the integrity of the traditional academic environment while encouraging the adoption of modern platforms such as online learning, mobile technologies, and cloud-based resources. This approach serves as a bridge between old and new educational methodologies, leveraging technology to enrich the learning experience while preserving the essential elements of traditional instruction [38].

The application of blended instruction has rapidly increased as educators recognize that varied delivery methods can boost student satisfaction and learning outcomes [54]. One significant advantage of blended learning is its ability to address some of the inherent disadvantages of purely e-learning environments. E-learning can inhibit the socialization process, leading to a lack of face-to-face communication [54]. Blended learning mitigates these issues by combining the strengths of traditional classes with the convenience of online courses. By integrating F2F interactions with online components, blended learning fosters socialization and enhances the learning process, acknowledging the human need for interaction. Furthermore, instructors can support their courses with online exercises, instant feedback, and enriched learning environments through hypermedia and multimedia, creating a more comprehensive educational experience [54].

2.4. Learning Objects

Traditional educational materials, such as lecture notes and exercises, are typically linear and hierarchical, structured in chapters or units of increasing difficulty. When these materials are transformed into a hypermedia format, the linear structure is often lost, resulting in discrete informational units. These digital units can be recombined to form coherent courses, which can be accessed and navigated through an online interface. These units are referred to as learning objects (LOs) [49]. LOs have emerged as a key concept in the realm of digital education, promising to revolutionize the way educational content is created, shared, and utilized. Defined broadly, a learning object is the smallest unit of educational content that holds intrinsic educational value [24]. Examples range from a simple graphic or text to more complex animations, videos, and interactive simulations [42]. This granular approach to content creation allows for flexibility and reuse, but it also brings challenges that need careful consideration.

The concept of learning objects draws inspiration from object-oriented programming, where the idea is to create modular, interchangeable units that can be combined to build more complex systems. A popular metaphor likens LOs to LEGO blocks, suggesting that they can be easily snapped together to build larger educational experiences [24, 39]. However, this metaphor is limited because not all learning objects can be seamlessly combined. A more fitting analogy might be that of atoms, which can combine in specific ways to form molecules, indicating that the combination of learning objects requires a thoughtful approach to maintain educational integrity and effectiveness [24].

One major challenge in the adoption and implementation of learning objects is the gap between the learning technology community and the education community. Terms such as "learning resource" or "learning object" are not native to educators, who are generally more focused on pedagogical outcomes rather than the technical properties or reuse potential of digital content. To bridge this gap, a clear demarcation of responsibilities is necessary. Developers should focus on the technical infrastructure, while educators concentrate on the pedagogical aspects. The development and use of learning objects can be viewed through a two-tier courseware development workflow, involving both production and consumption. In this model, developers create rendering software that educators can use to develop content. This separation of roles allows for a more efficient production process, where educators can choose from a range of available tools or develop their own if they have the necessary expertise [24].

Despite the theoretical benefits, the practical implementation of learning objects presents several challenges. The variability in quality and the lack of consistent classification schemes can make it difficult to find and integrate suitable objects into courses. Additionally, the promise of interoperability and reusability often falls short due to inconsistencies in size, format, and language of the learning objects. Addressing these challenges requires adherence to best practices in instructional design and a robust infrastructure for learning object repositories (LORs). Effective LORs should not only catalog learning objects but also include evaluations and usage information to aid educators in selecting the most appropriate resources [39]. The IEEE's Learning Objects Metadata (LOM) standard is widely used in higher education [31]. It aims to facilitate the identification, retrieval, and reuse of learning objects by providing detailed metadata. These metadata categories cover various aspects of the learning object, including technical properties, educational context, and pedagogical characteristics [49].

Learning objects that are published with an open licence fall under the category of Open Educational Resources (OERs) which are freely available for use, modification, and distribution. OERs play an important role in making educational content widely accessible. To achieve this, they must be easily searchable and discoverable, necessitating standardized metadata descriptions like LOM [31].

While learning objects hold significant promise for enhancing digital education, their successful implementation depends on overcoming various technical and pedagogical challenges. The complexity of educational problems cannot be resolved by technology alone. Educators often find that learning objects do not fit their specific course needs or are outdated. The promise of universal access and collaboration through learning objects assumes a level of adaptability and standardization that may not exist. As such, while the concept of learning objects holds substantial promise, realizing their full potential requires addressing these practical challenges and aligning them more closely with instructional design principles and educational goals [46, 55].

2.5. Learning Designer

The Learning Designer [30] is a web-based tool designed to assist educators in the systematic creation and sharing of learning designs, often conceptualized as detailed lesson plans. It facilitates the integration of learning technology into educational practices, thereby enhancing both teaching and learning processes. The platform offers a structured interface where educators can plan and display sequences of activities.

Central to the Learning Designer's functionality is its support for educators in crafting effective learning experiences. Users are invited to articulate teaching aims and outcomes, which can be categorized according to Bloom's taxonomy of educational objectives. This structured approach ensures that all learning activities are aligned with desired educational goals. The Learnign Designer offers six distinct learning types that can be specified for each activity:

acquisition (Read/Watch/Listen), inquiry, practice, production, discussion, and collaboration. These categories are based on Laurillard's Conversational Framework [32], which emphasizes iterative cycles of communication between educators and learners, as well as among learners themselves.

The tool's interface allows for detailed specification of each activity, including group size and duration, whether the activity occurs in the classroom or remotely. This comprehensive view helps educators consider the entire learning experience, ensuring a balanced and manageable workload for students, which is crucial for maintaining their motivation.

A feature of the Learning Designer is its dynamic pie-chart, which visually represents the overall learning experience being created. This analytical tool enables educators to quickly assess whether the balance of learning types aligns with their pedagogical intentions. Adjustments can be made by changing the type of activity, its duration, group size, or associated resources. This capability supports an iterative, reflective design approach, allowing for continuous refinement and improvement of learning designs.

The Learning Designer also promotes the effective use of technology in education by explicitly identifying the type of learning required, thus aiding educators in selecting appropriate digital tools. For instance, wikis or shared documents are suitable for collaborative production activities, while forums or blog comments facilitate discussions. The platform allows users to attach links to digital tools or Open Educational Resources (OERs) from anywhere on the web, enhancing the resourcefulness and interactivity of learning activities (Section 2.4).

Community building is another significant feature of the Learning Designer. It fosters a collaborative environment where educators can share their lesson plans. Each user has a personal space for saving their designs, which can be moved to a public space for others to view and adapt. Additionally, designs can be exported as MS Word documents for further editing, printing, or sharing.

3. Related Work

In the field of e-learning various approaches to developing e-learning courses have been proposed. This chapter compares eight approaches that were proposed between 2002 and 2013. Each approach is categorized by its goal, development process, focus on pedagogical design, collaboration, evaluation method, and involved stakeholders (Tables 3.1, 3.2, 3.3).

Montilva et al. [36] proposed a domain-specific approach for developing educational websites, drawing heavily from software engineering principles. Their approach emphasizes the systematic nature of course development, ensuring a robust framework for creating educational content. Between 2005 and 2007, there was a surge of approaches focusing on learning objects. In 2005, McDonald et al. [35] presented a three-stage approach to the collaborative development of learning objects. This method emphasized the importance of collaboration among educators to create effective and reusable learning materials. During the same period, Krauss and Ally [29] also explored the design and evaluation of learning objects in 2005. They identified various challenges and issues that arise during the creation of these objects, highlighting the need for careful consideration of educational design and assessment. Khan and Joshi introduced the people-process-product continuum (P3 model) for developing e-learning courses in their papers published in 2004 [27] and 2006 [28]. Using the P3 model the paper outlines a seven-phase development approach for e-learning courses, considering them from process, people, and product perspectives. In 2007, Hadjerrouit [19] proposed a nine-step development approach for the systematic creation of e-learning courses. This approach was designed to address problems stemming from inadequate requirements analysis and insufficient attention to pedagogical needs. By focusing on these critical areas, their method seeks to improve the quality and effectiveness of e-learning courses. Hadjerrouit expanded on his ideas in 2010 by defining a user-centered approach to develop and evaluate web-based learning resources (WBLRs) [20]. Hadjerrouit argues that educators and learners are not sufficiently represented during the development of WBLRs. With his four-stage approach he helps developers to translate technical and pedagogical requirements into a usable system that supports effective learning. Nedeva and Dineva described the stages of the design and development of e-learning coruses and their content in 2013 [40].

All the approaches utilize an iterative process model, employing prototypes that are incrementally refined and improved. This iterative refinement ensures continuous enhancement and alignment with educational goals and user needs. Despite their common use of iterative processes, the presented approaches define different scopes and phases, reflecting their unique focuses and methodologies. Montilva et al. divide their approach into a management process and a development process. The management process includes activities for planning, organizing, and controlling the project's success. Similarly, the approach by Nedeva and Dineva

Paper	Goal	Development Process
Nedeva and Dineva (2013) [40]	present the stages of the designing and development of e-learning courses and their content	iterative process create development plan design stage development stage review stage deployment stage
Hadjerrouit (2010) [20]	create a user-centered approach to develop and evaluate web-based learning resources in school education	iterative process with prototyping analysis design implementation and testing evaluation
Hadjerrouit (2007) [19]	create e-learning courses translate educational requirements	iterative process with prototyping system scope requirements determination requirements specification architecture design user interface design implementation delivery and use pedagogical evaluation
Khan (2004) [27], Khan and Joshi (2006) [28]	envisage entire development process in a modular approach using P3 model	iterative process planning design production evaluation delivery and maintenance instruction marketing
Krauss (2005) [29]	design and evaluation of learning objects describe challenges and issues associated with creating learning objects	iterative process with prototyping
MacDonald et al. (2005) [35]	three-stage process for the collaborative development of learning objects	paper-based document tailored for face-to-face classes repurposing of paper document into an electronic resource suitable for online courses creation of vibrant, rich, and interactive online learning object
Montilva et al. (2002) [36]	domain-specific process for developing instructional websites	iterative process with prototyping plan, organize and control project in management process instructional and technological activities in development process

Table 3.1.: E-learning development approach comparison (Goal, Development Process).

Paper	Roles	Collaboration
Nedeva and Dineva (2013) [40]	instructional designer project manager course developer content provider graphic designer reviewer technical specialist	not explicitly covered
Hadjerrouit (2010) [20]	educator developer student	distributed collaboration by means of e-mail, phone, and meetings educator-developer collaboration in analysis stage and evaluation stage
Hadjerrouit (2007) [19]	educator developer student	Collaboration through artifacts like Data Flow Diagrams and UML diagrams
Khan (2004) [27], Khan and Joshi (2006) [28]	content expert instructional designer developer graphic artist project manager director	collaboration in production phase participants interact with each other on a regular basis
Krauss (2005) [29]	instructional designer developer media designer subject matter expert	not explicitly covered
MacDonald et al. (2005) [35]	educator developer	extensive collaboration among team members in third stage developer role not highlighted
Montilva et al. (2002) [36]	educator developer	not explicitly covered

Table 3.2.: E-learning development approach comparison (Roles, Collaboration).

Paper	Pedagogical Design	Evaluation Method
Nedeva and Dineva (2013) [40]	determine learning objectives to formulate course content use of the pedagogical and methodological principles	none
Hadjerrouit (2010) [20]	involve students and teachers early in process focus on pedagogical principles and pedagogical usability evaluate pedagogical value and learning effect	comparison to other methodologies uses Montilva et al. comparison framework
Hadjerrouit (2007) [19]	blend of learning theories evaluate pedagogical value and learning effect	case study survey questionnaires, discussions with students and educator observations
Khan (2004) [27], Khan and Joshi (2006) [28]	comprehensive learner analysis pedagogically sound project plan	none
Krauss (2005) [29]	blend of learning theories define instructional problem before thinking about technology	case study
MacDonald et al. (2005) [35]	involvement of educators with diverse teaching backgrounds	grounded in practical experiences and supported by the research literature
Montilva et al. (2002) [36]	analysis of course's domain define learning objectives analyze students prior knowledge, skills, learned competences	comparison to other methodologies defines comparison framework

Table 3.3.: E-learning development approach comparison (Pedagogical Design, Evaluation Method).

includes a phase for creating a development plan, which clarifies expectations, schedules, and manages resources and costs. The phases in these approaches exhibit significant overlap, though some, like those by Montilva et al. and Hadjerrouit, are more detailed. For instance, Hadjerrouit's phases of System Scope, Requirements Determination, and Requirements Specification can be combined into an Analysis phase. Similarly, Architecture Design and User Interface Design can be merged into a Design phase. Montilva et al.'s phases of Course Analysis, Requirements Definition, and Quality Attribute Analysis can also be consolidated into an Analysis phase. By adopting these adjustments, the following phases can be identified across the approaches:

Planning Includes project management activities such as project planning and control, team organization, staffing, and direction. Additionally, this phase entails quality control as well as selecting and acquiring course-creation tools and providing training for the development team and users [36, 40].

Analyze Involves defining the system scope, gathering requirements, analyzing the course domain, and defining requirements [36, 19]. During this phase the needs and capabilities of learners are understood and reviewed for instructional and pedagogical soundness [28]. This can be achieved by defining learning objectives [29] or planning the course as if it were a traditional lesson [35].

Design Course content is formulated, and appropriate educational methods are selected. Furthermore, the user interface is designed. Content is analyzed and organized into logical course structure objects. Additionally, appropriate delivery mediums are selected based on pedagogical principles [40, 19, 36, 20].

Development Create vibrant, rich, and interactive courses or learning objects [35]. It includes activities such as creating a page layout, multimedia development, interactive content design, and quiz and test development [40, 36]. Reusing learning objects and systematic testing are also part of the development phase [20].

Delivery and Maintenance Deploy and use the prototype, ensuring that the course is functional and accessible to learners [19].

Evaluation Assess the quality of the final product. Pedagogical evaluation is performed [19], and content approval is sought [40]. Additionally, the development process and final product are evaluated to ensure they meet the desired standards [20].

Khan and Joshi have defined a marketing phase, which is not considered in this thesis' context of e-learning course development. Excluding this phase, the identified phases align with those of the ADDIE model [7, 23]. However, there is a deviation from ADDIE in the inclusion of the planning phase. This phase is exclusively for project management and does not directly pertain to instructional design. Nonetheless, it is crucial for development approaches involving multiple participants, ensuring that the project is well-coordinated and that all stakeholders are aligned.

The involved roles vary slightly across the different approaches. The involvement of educators and developers is a common thread. This highlights the critical importance of educator involvement in the development process, as they understand how to structure content effectively and are familiar with their students' needs. Other roles include students, instructional designers, project managers, graphic artists, content experts, media designers, and reviewers.

Although all approaches consistently involve educators and developers, the collaboration between these roles is not always detailed. Hadjerrouit [20] identifies email, phone, and meetings as primary communication methods among participants [20]. Collaboration typically occurs during the analysis and evaluation phases. In Hadjerrouit's approach [19], collaboration is implied through artifacts such as Data Flow Diagrams and UML diagrams, although these interactions are not elaborated upon. Khan et al. [27, 28] emphasizes that individuals involved in various stages of the e-learning process should maintain regular contact with each other to ensure cohesive development [27, 28]. McDonald et al. specifically address the collaboration among educators, highlighting how different expertise areas contribute to creating a comprehensive and effective learning object. However, the role of developers is acknowledged but not explored in depth [35]. In contrast, Montilva et al., Krauss and Ally, and Nedeva and Dineva do not discuss collaboration at all [36, 29, 40].

All authors agree that effective e-learning courses can only be created by focusing on pedagogical design and leveraging technology to achieve outcomes that are otherwise not possible. A common method to specify the goals of an e-learning course is through clearly defined learning objectives. These objectives are instrumental in aligning the efforts of participants and in selecting appropriate educational methods [36, 29, 19, 20, 40].

The formulation of course content is driven by these educational goals. Nedeva and Dineva emphasize the importance of aligning course content with the established learning objectives to ensure that the educational outcomes are met effectively [40]. Additionally, Khan [27] and Montilva et al. [36] highlight the need for a comprehensive learner analysis to tailor the course to the specific knowledge, skills, and competences of the students. This includes an analysis of students' prior knowledge and an understanding of the course domain.

McDonald et al. focus on pedagogical design by heavily involving educators with diverse backgrounds in the development process. This approach ensures that various perspectives are considered, enriching the educational content and delivery methods [35]. Furthermore, learners are integrated into the process to provide feedback and participate in the evaluation of the course, as highlighted by Hadjerrouit [20].

A blend of learning theories is often used in the pedagogical design of e-learning courses. Krauss and Ally [29] and Hadjerrouit [19] advocate for the integration of multiple learning theories to cater to different learning styles and needs. This theoretical foundation supports the creation of a robust and flexible learning environment.

Hadjerrouit [19] specifically introduces a pedagogical evaluation phase to assess the effectiveness of the course design from an educational perspective. This phase ensures that the course not only meets technical standards but also achieves its educational objectives.

Evaluation methods across different approaches are quite similar, predominantly employing two main methods. Case studies, used by Krauss and Ally [29] and Hadjerrouit [19], involve developing a course using their specific approach. The resulting course or learning object is then used to instruct students. Afterwards, these studies employ survey questionnaires and interviews to gather feedback. Additionally, Hadjerrouit [19] includes educators and student observations in the evaluation process. Montilva et al. developed a comparison framework to evaluate their approach against other methodologies available in the literature. This method was also adopted by Hadjerrouit [20]. This framework-based evaluation is less resource-intensive as it does not require course development, focusing instead on methodological comparison [36, 20].

In summary, while the approaches to developing e-learning courses share many similarities, especially in the phases and evaluation methods, the collaboration between different roles, such as educators and developers, remains underexplored. These roles are pivotal in the e-learning development process, bringing together technical and pedagogical expertise. The phases of the approaches align closely with the ADDIE model, indicating their proven effectiveness and reducing the need for redefining these phases. The analysis phase is particularly crucial, laying the groundwork for the subsequent development of the e-learning course. The subsequent chapter will build on these insights to describe a new development approach that emphasizes collaboration between educators and developers, aiming to enhance the overall effectiveness and quality of e-learning courses.

4. The Collaborative Development Approach

In this chapter, a development approach for e-learning courses that emphasizes the collaboration between educators and developers is described. Drawing from the insights and methodologies discussed in the previous chapters, this approach aims to address the existing gap in explicitly detailing the collaborative efforts required in creating effective e-learning courses. By focusing on the unique contributions of both educators and developers, the proposed approach seeks to enhance the quality and efficacy of e-learning courses through a structured, iterative process that fosters continuous interaction and feedback. The development approach is designed to integrate the pedagogical expertise of educators with the technical skills of developers, ensuring that educational content is both pedagogically sound and technically robust.

4.1. Roles

The collaboration between educators and developers is critical for the success of e-learning courses. Educators bring essential pedagogical knowledge and understanding of student needs, while developers provide technical expertise and innovative solutions. When these two roles closely work together, they can create more effective and engaging learning experiences. This collaboration ensures that educational content is not only technically robust but also pedagogically sound (Section 2.2). This section introduces the roles of educators and developers in this approach. The description of the roles is not intended to be exhaustive, as there is no universally accepted definition. The aim is to provide a foundational definition for the following sections based on common tasks each role performs. The tasks are derived from descriptions in the related work. Future improvements and extensions may consider additional roles that are found in the literature (Chapter 3).

4.1.1. Educators

Educators are considered to be the main drivers of development because they want to use the created e-learning course for their instruction. They have detailed knowledge about their target audience and possess a pedagogical background. It is assumed that they have limited time for meetings as they already have a heavy workload [20]. In addition, educators might have limited technical understanding and might lack experience with e-learning. In the context of e-learning course creation, the role of educators encompasses several key responsibilities (Figure 4.1).

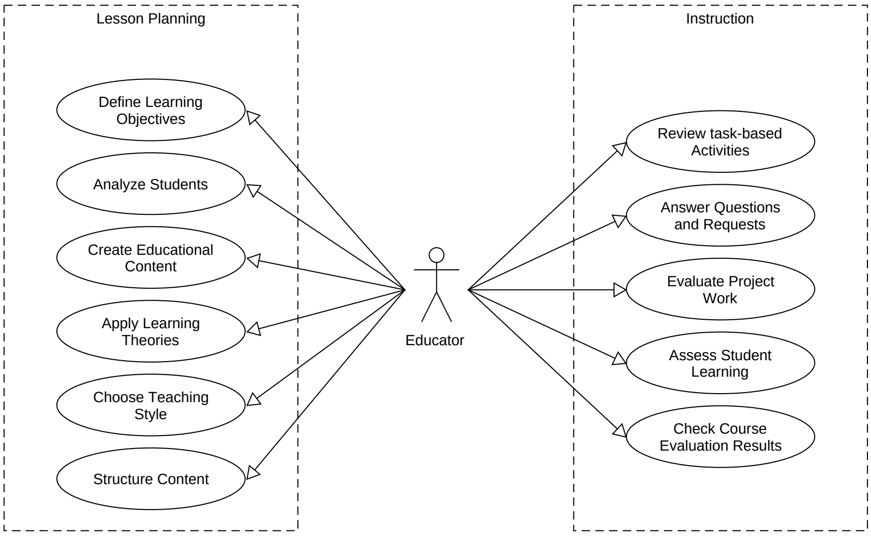

Figure 4.1.: Educator responsibilities

During the e-learning course creation, educators are responsible for creating educational content that aligns with learning objectives. The content must be structured in a coherent and logical flow of information. This involves organizing content into manageable teaching units and structuring the individual lessons. Educators are able to apply learning theories (Section 2.1) and adapt their teaching style to the learners' characteristics, learning styles and prior knowledge. They assess whether the complete lesson should be facilitated through an e-learning course or if a blended learning design is better suited to achieve their pedagogical goals.

During instruction, educators are responsible for guiding and overseeing the students' learning process. In a traditional classroom setting, educators often take on a central role. However, in e-learning courses, the focus shifts more towards the students, allowing them to set their own learning pace. In this context, the educator adopts a supervisory role. They review completed assignments, respond to questions, and collect feedback to make necessary course adjustments.

4.1.2. Developers

Developers play a crucial role during the development of e-learning courses (Figure 4.2). They may not have a connection to the subject matter being taught and lack pedagogical expertise. It is not guaranteed that developers are available during the complete development process. Developers possess varying levels of technical expertise and may not have experience with e-learning.

Their expertise lies in analyzing stakeholder requirements and converting them into practical software solutions. Typical development phases involve architectural design, user interface

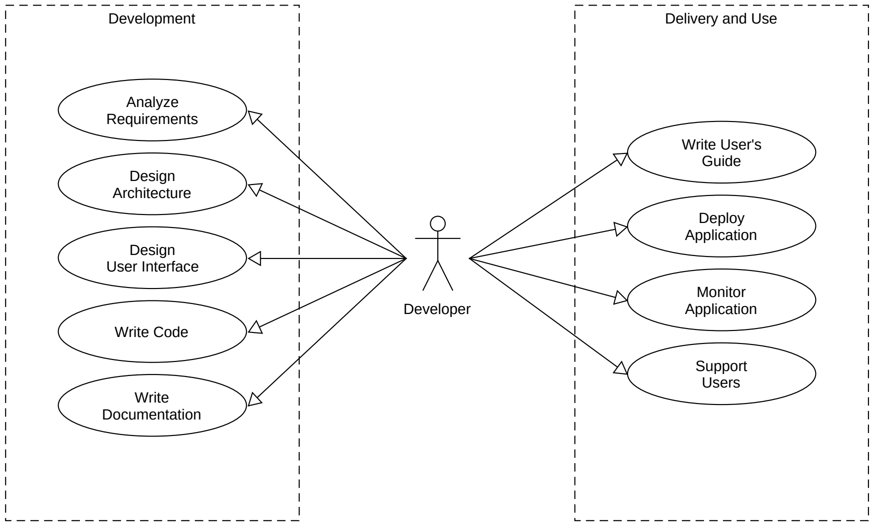

Figure 4.2.: Developer responsibilities

design, writing code, testing, and creation of documentation. In collaboration with educators, these activities help that e-learning courses meet educational goals effectively and function reliably for users.

Once an e-learning course is developed, developers are responsible for deploying it for use by educators in their instructional activities. They ensure the smooth implementation of e-learning courses and extend support through application monitoring and user assistance in case of any issues. Furthermore, developers may create user guides to enhance user understanding and facilitate effective utilization of the e-learning course.

4.2. Inspiration from Software Product Line Engineering

The idea of software product lines has been around since the 1960s and received a lot of attention in the 1990s. This approach to software development argues that instead of developing software systems from scratch, they should be constructed from reusable parts. This approach allows software to be tailored to the specific requirements of the customer, who can select from a vast array of configuration options. The software product line approach provides a form of mass customization by constructing individual solutions based on a portfolio of reusable software components. This method introduces individualism into software production while retaining the benefits of mass production, allowing entire market segments to be effectively served. The need for this individualism arises from the diverse requirements for software functionality, target platforms, and quality properties such as performance and energy consumption [3].

The software product line development approach offers several significant benefits. First, it allows for tailor-made solutions, enabling developers to produce a wide range of customized products instead of providing a single standardized product or a few preconfigured editions.

	Problem Space	Solution Space
Domain Engineering	Domain Analysis	Domain Implementation
Application Engineering	Requirements Analysis	Product Derivation

Table 4.1.: Task clusters in product line development [3].

Second, it reduces costs as vendors do not have to design and develop each product from scratch. Instead, they create reusable components that can be combined in various ways. While the initial investment in designing these reusable parts is higher, it pays off in the long term by significantly reducing the development cost per product. Third, software product lines improve quality. Similar to industrial mass production, standardized software components can be systematically checked and tested across multiple products, leading to more stable, lean, and reliable software. Lastly, this approach shortens the time to market. By using predefined configuration options and existing parts, vendors can quickly assemble the desired product. Even when new functionality is needed, building on top of well-designed, reusable components is much faster than starting from scratch. This ability to rapidly respond to market changes offers a significant advantage over both standardized and individually developed software [3].

The concept of a domain in software product line development underscores the commonalities shared among different software products within a specific scope. The breadth of this domain significantly influences the potential for reuse. When the domain is narrowly defined, its applicability is limited to a small subset of stakeholders. Conversely, an overly broad domain results in products that diverge too greatly to effectively reuse components [3].

In software product lines there is a separation between domain engineering and application engineering, and between problem space and solution space (Figure 4.1). Domain Engineering involves analyzing the domain of the product line and creating reusable artifacts. It does not produce a specific software product but prepares artifacts that can be used across software products. This phase targets development for reuse. Application Engineering focuses on developing specific software products to meet the needs of stakeholders. It reuses artifacts from domain engineering and corresponds to traditional software product development but with an emphasis on reuse. This phase is repeated for each software product derived from the product line. Problem Space reflects the perspective of stakeholders, capturing their problems, requirements, and views of the domain and individual software products. Solution Space represents the developer's perspectives, involving the design, implementation, validation, and verification of features and their combinations to facilitate systematic reuse [3].

These distinctions give rise to four clusters of tasks in product line development:

Domain Analysis This is a form of requirements engineering for the entire product line. It involves determining the domain scope, identifying which features should be covered, and which features should be implemented as reusable artifacts.

Requirements Analysis Part of application engineering, this investigates the needs of specific stakeholders. Requirements are mapped to feature selections based on domain analysis.

Novel requirements may feed back into domain analysis, potentially modifying the feature model.

Domain Implementation This involves developing reusable artifacts corresponding to features identified in domain analysis.

Product Derivation This is the production step in application engineering, where reusable artifacts are combined according to the requirements analysis. This process can be more or less automated and may involve various development and customization tasks.

Domain engineering is performed once for the entire product line, while application engineering is done for each individual software product. The goal of product line development is to shift as much development effort as possible from application engineering to domain engineering. For example, if quality assurance can be done in domain engineering instead of for individual software products, costs can be significantly reduced. Shared artifacts are investigated only once, making the overall development more efficient and cost-effective [3].

A similar approach could prove beneficial in e-learning course development because traditional e-learning course development, focuses on creating a single course. Educators and developers often collaborate to create e-learning courses. The resulting course is tailored to the specific needs and preferences of the involved educators. However, different educators who wish to reuse the same course might struggle because it was customized for the original context and individuals. Even though the topic covered by the course remains the same, different educators will likely want to create their own version of the e-learning course to cater to their unique preferences. Educators might want to emphasize different aspects of the subject matter or tailor the content to their students' varying levels of prior knowledge, learning styles, and interests. Additionally, educators may need to conform to the rules and guidelines of their institution, including design standards, licenses, and copyright regulations. Furthermore, educators might wish to add their own content and expand the course to include additional material or perspectives. This need for customization requires a shift in the development approach to make courses more adaptable and reusable across different contexts and educator preferences. Instead of focusing on one course, a variety of similar but distinct courses should be considered.

In the realm of e-learning, a lesson serves as a suitable domain because courses centered on the same lesson topic usually exhibit similarities. For example, in a lesson covering integer addition, sections with examples and practice tasks are standard components. Additionally, components like math equation rendering tools can enhance reuse across various math-related courses.

4.3. Reusable Building Blocks

The idea of creating reusable artifacts which can be used in multiple courses is not new. Reusability is a desirable characteristic of software artifacts, especially in an era where software projects are continually growing in complexity. To meet the demands for high quality, reasonable cost, and a short time to market, it is crucial to streamline the development process. Software reuse is believed to be one such approach to achieve these goals. However, the practice of reuse has proven to be complex, and there are many misconceptions about how to implement and gain benefits from it [2]. When implemented correctly, reuse can significantly reduce development and maintenance costs. As early as 1968, McIllroy proposed at a NATO Software Engineering Conference an industry of off-the-shelf, standard source-code components and envisioned the construction of complex systems from small building blocks available through catalogs [2]. The final objective is to make something once and reuse it several times. The most common form of reusable artifact is source code, but it is not the only one. The idea involves reusing requirements specifications, design, architecture, test data, and documentation [2].

In the context of e-learning, the concept of reusability can be found in the form of learning objects (Section 2.4). Learning objects are modular, digital resources that can be used and reused to support learning across different contexts. These can include text, images, videos, quizzes, and interactive activities. By designing learning materials as reusable objects, educators can create a more efficient and flexible educational experience. For example, a learning object that explains a fundamental concept in mathematics can be reused across different courses, saving time and effort in content creation as well as software development. This approach also ensures consistency in the quality and delivery of educational content. Moreover, learning objects can be customized and adapted to fit specific learning needs, making them versatile tools in e-learning.

However, the reuse of traditional learning objects faces notable limitations. While these objects offer a modular approach to content creation, they often lack the flexibility to meet specific educational needs due to their one-size-fits-all approach. The reusability paradox highlights the tension between the desire to create highly reusable learning objects and the need for these objects to be pedagogically effective and context-specific. The paradox can be summarized as follows: the more context-specific a learning object is, the more effective it tends to be in its intended educational setting. However, this specificity reduces its reusability in different contexts. Conversely, the more generic a learning object is, the more reusable it becomes, but it often loses pedagogical effectiveness because it cannot cater to specific learning needs and contexts [29]. This paradox presents a significant challenge in the design and implementation of learning objects. On the one hand, educational institutions and developers aim to create content that can be widely reused to maximize resource efficiency and reduce costs. On the other hand, educators seek content that is tailored to their specific teaching objectives and the unique needs of their students.

LMS like Moodle offer their own version of reusability as predefined activities and resources. Resources are static elements like text, pages or URLs. Activities require student participation like assignments, chat, choices, feedback, forum or quizzes. Furthermore, Moodle offers the ability to include H5P elements [18] which allows the reuse of existing HTML5 content and applications [37, 48]. These predefined building blocks allow educators to create their courses in a modular way with limited exposure to technical details. This is very desireable as it reduces the complexity of the development and puts educators in the center of the course creation process.

Nonetheless, this approach is only powerful if the offered building blocks completely satisfy the educator requirements. LMSs tend to be largely generic [41]. This puts a limit on the creativity of the educators and forces educators to make compromises because of technological restrictions. This is contrary to the definition of e-learning introduced in Section 2.2 which states that e-learning is pedagogy empowered by technology. By inlcuding developers in the development process, educators are able to request bespoke building blocks that better align with their instructional objectives and student needs. This customized approach enables a more dynamic and responsive educational environment. Educators identify gaps or limitations in existing building blocks and collaborate with developers to create new functionalities or customize existing ones. Together they outline their specific requirements which are translated into functional, interactive building blocks. This process not only enhances the quality of the educational content but also ensures that it is directly relevant to the changing needs of educators and learners. This collaborative approach promotes continuous improvement. As educators implement and use these custom-built building blocks, they can provide feedback to developers, who can then refine and enhance them. This iterative cycle of development and feedback ensures that the e-learning platform remains responsive and effective.

4.4. Phases

This section outlines the phases of the collaborative development approach, which are derived from the distinctions between problem space, solution space, domain engineering, and application engineering discussed in the software product line development section (Section 4.2). These phases have been renamed to align more closely with the terminology used in e-learning. In addition to these four core phases an instruction phase is defined (Table 4.3). The following sections provide a detailed description of each phase, highlighting common tasks and interactions with other phases.

4.4.1. Lesson Planning

The Lesson Planning phase is dedicated to the pedagogical design of a lesson, which is a crucial component of effective e-learning courses (Section 2.2). This phase is integral to the e-learning course development approach and corresponds to the Analyze phase of the ADDIE model commonly used in other development frameworks (Chapter 3).

As the entry point into the development of a new e-learning course, the Lesson Planning phase focuses exclusively on pedagogical design. The lesson is conceptualized as if it were to be delivered in a traditional classroom setting, eliminating technical details and placing emphasis on pedagogy. This approach allows educators to utilize their domain expertise and perform this phase independently from developers. Positioned within the problem space and domain engineering, this phase lays the groundwork for future e-learning courses that will benefit from the pedagogical considerations established here.

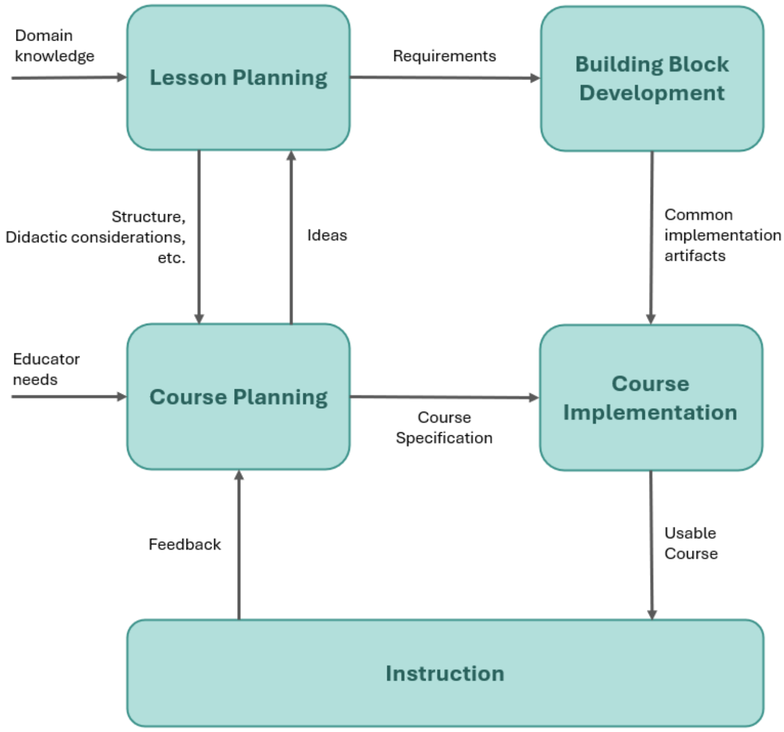

Figure 4.3.: Collaborative Development Approach Phases based on [3]

Planning begins with defining the topics to be covered, the target group of learners, the schedule, and any prerequisites for the lesson. The lesson is divided into teaching units, typically corresponding to individual classroom sessions. For each teaching unit, learning objectives are established to guide lesson planning and ensure alignment with desired educational outcomes. A rough content analysis is conducted to outline key concepts of the topics. Additionally, didactic considerations and a comprehensive learner analysis are performed to tailor the lesson to the target group's needs and characteristics, enhancing instructional effectiveness. Documentation also includes acquired competencies, curriculum alignment, framework conditions, and instructional methods.

Each teaching unit is further segmented into teaching phases, which detail the planned classroom activities and their time frames. These phases are placed within the learning cycle according to the phases defined in the Learning Designer (Section 2.5). The educator's presence — whether physically present, available online, or not available — is documented to determine if a course requires a blended learning design (Section 2.3). This thorough documentation ensures that each teaching unit is aligned with overall educational goals.

The plans and considerations developed during this phase are valuable across multiple courses covering the same topic, providing a consistent framework adaptable to various contexts during the Course Planning phase. While some educators might find the extensive documentation unnecessary, particularly if they repeat the same lesson annually and internalize many considerations, detailed documentation is crucial. It enables developers to effectively contribute and comprehend the lesson's pedagogical design. Educators aiming to reuse the lesson or contribute to it also benefit from thorough documentation.

The lesson plan is continually refined and updated based on feedback and insights from the derived courses, ensuring it remains relevant and effective. This iterative process provides a robust foundation for future e-learning courses. Additionally, the interaction during this phase identifies the need for reusable building blocks, which triggers the Building Block Development phase to meet the identified requirements and use cases.

4.4.2. Course Planning

The Course Planning phase is dedicated to translating a lesson plan into a functional and effective e-learning course. Building upon the pedagogical foundation established during the Lesson Planning phase, this stage ensures that educational design remains central throughout the development process. This phase aligns with the Design phase of the ADDIE model, as discussed in related work. The courses are based on the requirements of individual educators. Thus, it is possible that two educators create two distinct courses based on the same lesson plan reflecting their context and preferences.

During this phase, close collaboration between educators and developers is vital. Educators present their vision for the course, including the desired learning outcomes, instructional strategies, and content flow. Developers assist by translating these ideas into a technical specification, ensuring that the course is both pedagogically sound and technically feasible. Tailoring the planned lesson to fit the specific needs and style of an individual educator allows for smaller variations from the already planned instruction, making the course relevant and effective for the targeted teaching context.

The primary tasks in the Course Planning phase involve deciding which teaching units and teaching phases from the lesson plan should be included in the e-learning course. Additionally, defining the overall structure of the course is crucial. One approach is to organize the e-learning course into pages, which are intuitive for educators and can be easily mapped to the subsequent Course Implementation phase. User interface design is especially useful as a communication artifact, allowing educators to quickly sketch out their ideas without having to use modeling languages or write extensive text. Developers can then present their ideas or intermediate results to educators in an easily understandable format. Deciding on the technology to be used for course implementation is another key activity.

The detailed plans and considerations developed during the Course Planning phase provide a strong foundation for the subsequent Course Implementation phase, where the course will be brought to life. Course Planning and Course Implementation are performed in iterative cycles. The course is repeatedly reviewed and refined based on feedback, ensuring continuous improvement. Educators provide feedback on the course as it is being developed, and developers make adjustments based on this feedback.

4.4.3. Building Block Development

During the Building Block Development phase, reusable components that form the foundation of multiple e-learning courses are created. This phase ensures that the developed building blocks are both reusable and meet the specific educational needs of educators. It is triggered by requirements from lesson plans that should be encapsulated into a building block. These requirements may arise through ideas during the Course Planning phase in collaboration with developers. Such ideas are then incorporated into the lesson plan, resulting in a set of requirements for the building block. The development of building blocks encompasses all phases of the ADDIE model and is performed in parallel with the other phases. Building blocks are refined in iterative cycles until they are ready to be used during the Course Implementation phase.

During the Analysis phase, developers work closely with educators to refine the requirements. Developers must strike a balance between creating versatile building blocks and ensuring that these blocks can be tailored to specific educational contexts. This involves identifying the educational goals and objectives that the building blocks need to support. Through collaboration between educators and developers, technical and pedagogical requirements of the building block are documented. New requirements may surface in each iteration by including educators that work with different target groups and educational contexts.

In the Design phase, the requirements gathered in the Analysis phase are translated into detailed specifications for the building blocks. This includes defining the functionalities and features of each building block, creating detailed design documents that outline the structure and possible interactions, designing user interfaces that are intuitive for educators and engaging for learners, and planning for scalability and flexibility to ensure that the building blocks can be adapted for various courses. Educators may find it challenging to be involved in this phase as it is more focused on technical aspects. However, mockups of the user interface offer a possibility to involve educators. These mockups visualize how a developer imagines the final building block, usually omitting technical details and focusing on conveying the general idea. Educators should be able to comment on the mockups and provide feedback for developers. Conversely, educators may sketch their vision of the user interface to help developers understand their expectations and ideas.

The Development phase is where the actual creation of the building blocks takes place. This phase involves writing code for each building block and conducting initial testing to identify and fix any issues early in the development process.

In the Implementation phase, the building blocks are integrated into pilot courses to test their effectiveness in real educational settings. This phase involves deploying the building blocks in selected courses, training educators on how to use and customize the building blocks to fit their specific needs, and collecting feedback from both educators and learners on the usability and effectiveness of the building blocks.

During the Evaluation phase detailed feedback from educators and learners is gathered to assess how well the building blocks meet their intended educational goals. The results of the evaluation are fed back into the Analysis phase.

Throughout the Building Block Development phase, developers and educators maintain close collaboration. Educators provide insights into the pedagogical requirements and offer feedback on the usability of the building blocks. Developers focus on the technical aspects and ensure the components are robust and adaptable. The primary goal of this phase is to create building blocks that can be reused across multiple courses while still allowing for customization to meet specific educational needs. These reusable building blocks are used during the Course Implementation phase, where they will be assembled and customized to create e-learning courses.

4.4.4. Course Implementation

The Course Implementation phase combines the results of the previous phases to create a functional and engaging e-learning course. The course specification created during the Course Planning phase serves as the primary input. In combination with the lesson plan and the previously created building blocks, the final course is assembled. The pedagogical design of the course should guide the implementation. This phase maps to both the Development and Implementation phases of the ADDIE model.

To create the e-learning course, developers choose suitable building blocks, which were created in the Building Block Development phase, based on their ability to meet the specific course requirements. Building blocks may require various inputs such as texts, videos, images, and interactive elements. The content for the course is primarily provided by educators, who use their subject matter expertise to ensure that the material is pedagogically sound. Developers collaborate with educators to integrate multimedia elements and interactive features that enhance learner engagement and comprehension.

Initial testing involves deploying a prototype of the course to a small group of educators. This allows for practical evaluation and feedback, which is crucial for identifying areas for improvement. Based on the feedback gathered, necessary adjustments are made to refine the course content and functionality. This iterative process continues until the course meets the desired standards of quality and effectiveness to be used in Instruction.

Once the course is refined, it is prepared for deployment. Detailed deployment instructions and guidance are provided to educators, ensuring they are well-equipped to deliver the course. This includes creating user manuals, instructional guides, and necessary training to familiarize educators with the course's features and functionalities.

4.4.5. Instruction

The Instruction phase marks the transition from course development to real-world application. During this phase the finished e-learning course is actively used by educators to instruct their students. This puts the pedagogical and technical aspects of the course to the test in a real-world educational setting. Educators utilize the e-learning course to deliver lessons, facilitate activities, and engage students in the learning process. Feedback from both students and educators is a vital component of the Instruction phase. It helps to assess whether the course meets educational goals. Feedback can be collected through various methods such as surveys, interviews, and observations. Developers can support this phase by integrating monitoring and analytics features directly into the e-learning course. Thus, allowing for data collection on student interactions, progress, and performance. These analytics can highlight trends, pinpoint problematic areas, and suggest potential modifications. Once feedback is collected, it is analyzed and used to inform the next cycle of the Course Planning phase which starts a new iterative cylce of the development approach.

5. The Collaborative E-Learning Course Creation Platform

An e-learning course creation platform is developed to make the collaborative development approach introduced in Chapter 4 usable. This chapter details the development of the platform. First, the vision of the platform is described in Section 5.1. The extent of the platform is too large to be fully implemented during this thesis. Thus, the scope of implementation is defined in Section 5.2. Afterwards, the included features (Section 5.3) and the architecture (Section 5.4) are described. Finally, considerations about deployment are outlined in Section 5.5.

The source code and related documentation as well as the deployment scripts can be accessed through the KIT thesis repository (Chapter 8).

5.1. Vision

E-learning has revolutionized education by providing new avenues for delivering knowledge, yet the creation of effective e-learning courses remains a complex process requiring collaboration among various stakeholders, including educators, developers, students, instructional designers, and content creators. While existing approaches recognize the importance of this collaboration, there is a significant gap in operational guidance and tools to facilitate it effectively. This section envisions a collaborative e-learning course creation platform (ELST) designed to address these challenges by providing a seamless, integrated environment for all participants.

The primary audience for this platform includes individuals involved in the creation and continuous improvement of e-learning courses. Educators, who aspire to design engaging and dynamic courses, often find themselves hindered by technical details they are not equipped to handle. Developers, on the other hand, rely on input from educators to create courses that are both effective and sound in pedagogical design. Both groups seek to work independently yet collaboratively, iterating quickly and efficiently to enhance the learning experience.

To meet these needs, the envisioned platform will offer a range of features designed to streamline the course creation process (Figure 5.1). The design of the platform is guided by the collaborative development approach introduced in Chapter 4. Educators will be able to create and share detailed lesson plans with their peers. These lesson plans can then form the basis for e-learning courses. The platform will support both fully digital courses and blended learning designs, allowing educators to choose the format that best suits their instructional needs.

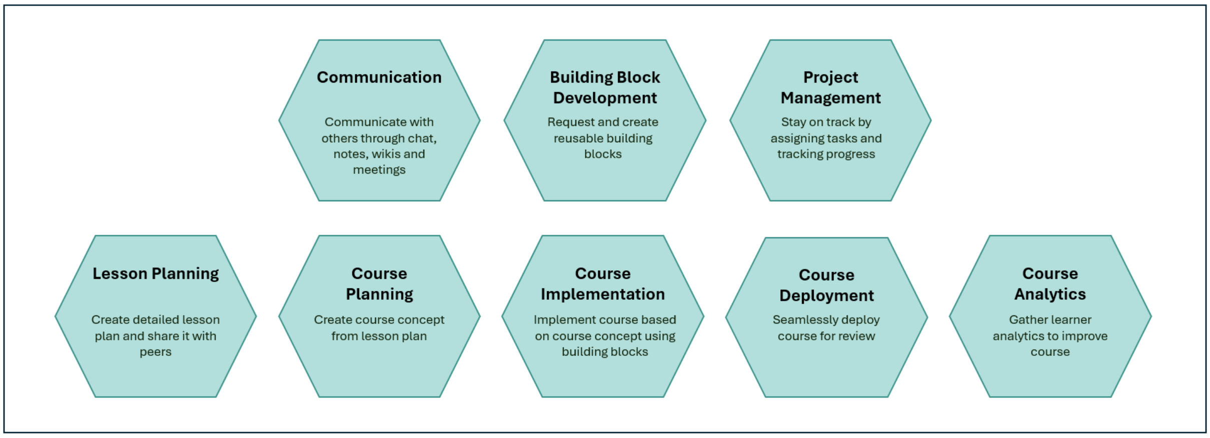

Figure 5.1.: ELST Features

Collaboration will be enhanced through integrated tools such as chat, notes, wikis, and virtual meetings facilitating communication and coordination among team members. Project management features will enable users to assign tasks, track progress, and ensure that deadlines are met. A central repository of reusable building blocks will improve implementation speed, allowing educators and developers to build courses efficiently using pre-designed components while also supporting individual customizations. It is conceivable that the course implementation can be performed through a semi-automated process that draws implementations from a central building block repository.

Seamless deployment features will enable courses to be reviewed and refined continuously, with built-in tools for evaluating the pedagogical design of both courses and individual building blocks. Additionally, the platform will gather learner analytics, providing valuable insights that can be used to further enhance the quality of the courses.

The goal is to provide an all-in-one platform for creating e-learning courses that feels intuitive and accommodating for all stakeholders. The unique value proposition of this platform lies in its ability to centralize and simplify the collaboration process for all stakeholders involved in e-learning course creation. By covering the complete development process — from lesson planning to deployment — it provides tailored views for each stakeholder, allowing them to focus on their specific tasks without being held back by unnecessary details.

5.2. Scope of Implementation

Given the extensive scope of the envisioned platform, it is necessary to limit the implementation to a subset of features (Figure 5.2). This focused approach ensures that the project remains manageable within the constraints of this thesis while allowing for valuable feedback from educators and developers to gauge the interest and validity of the proposed solution. Consequently, the initial development will concentrate on creating a Minimum Viable Product

(MVP) that includes core functionalities essential for validating the platform's approach and utility.

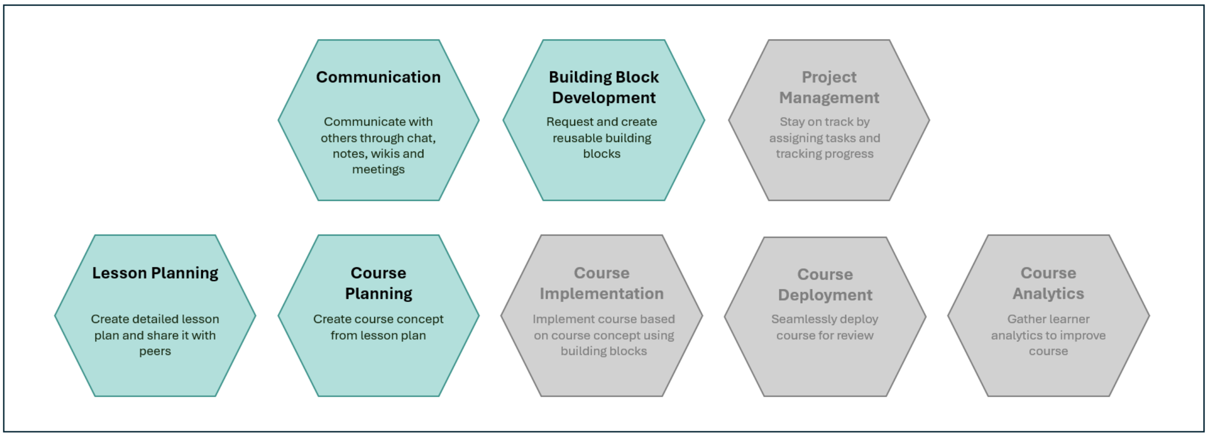

Figure 5.2.: Scoped ELST Features

The MVP will primarily enable educators to plan their lessons in a simplified manner. This lesson planning functionality will allow educators to create and share detailed lesson plans with their peers, forming the foundational content for subsequent course development.

In addition to lesson planning, the MVP will support basic course planning features. These features will allow educators and developers to translate lesson plans into coherent courses, facilitating a structured approach to e-learning course creation. To test the practicality and effectiveness of building blocks, the MVP will include a basic version of this feature. The initial implementation of this feature will enable educators to request building blocks from developers by providing an initial description. These building blocks will be defined solely by their metadata, without any concrete implementations associated with them. The building blocks can be reused across different courses and configured to fit to their context.

Communication features are a cross-cutting concern for all functionalities. They will be integrated into the MVP to support communication and coordination during the course planning and implementation phases. Integrated notes and chat functionalities will facilitate asynchronous communication between educators and developers.

By focusing on these core features, the MVP will provide a practical and functional subset of the envisioned platform. This approach allows for a thorough evaluation of key concepts and the collection of feedback from end-users. The insights gained from this initial implementation will be helpful in refining and expanding the vision of the platform in future development phases, ensuring that it meets the needs of all stakeholders involved in e-learning course creation.

5.3. Features

This section outlines the features implemented in the initial version of the platform, as described previously. Figure 5.3 illustrates the core concepts of the application and their interrelations. Before developing an e-learning course, educators must plan their lesson. The lesson plan encompasses pedagogical design considerations and provides a foundational structure. When an educator decides to create an e-learning course based on this plan, a new course concept is generated. This course concept specifies the technology to be used for the course implementation and defines the course structure, including mockups and various implementation notes. The course concept serves as a specification for the course implementation. Suitable building blocks are then selected from the building block repository. The repository contains building blocks requested during the course conceptualization phase. Only building blocks compatible with the desired course technology should be selected. In this initial version, the platform does not support developers in writing code. It is assumed that code for building blocks is written and hosted in separate version-controlled repositories. Currently, course deployment must also be performed outside the platform.

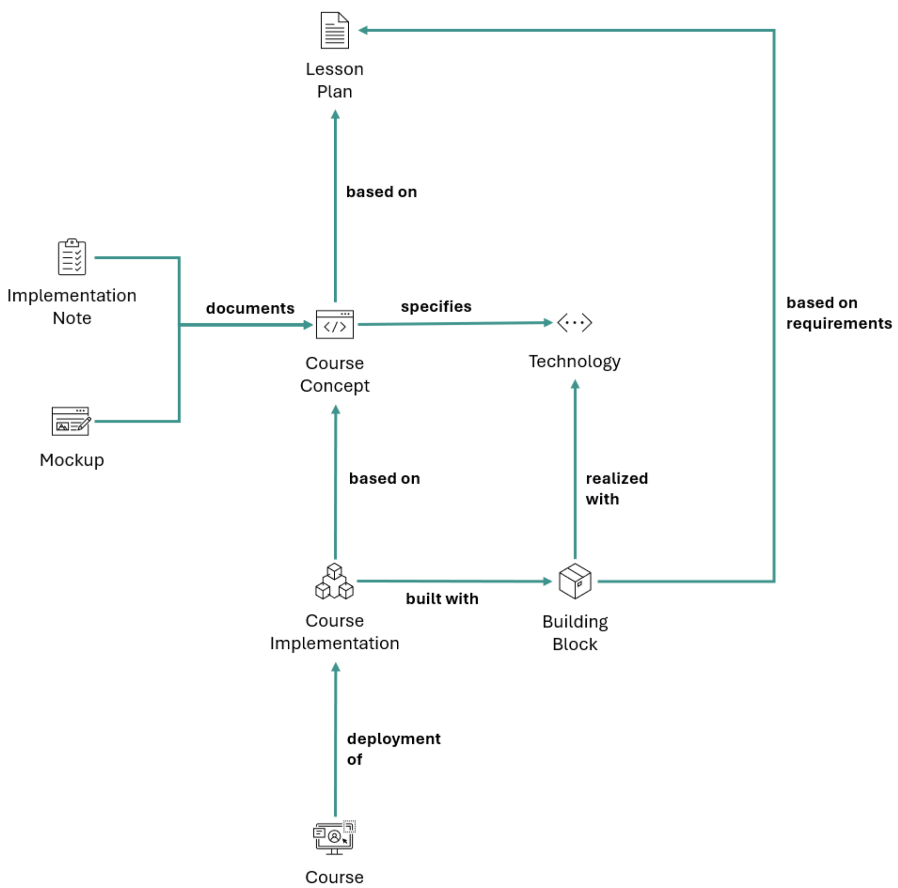

Figure 5.3.: Application Sketch

5.3.1. User Management

User management is a fundamental feature in many applications, providing a robust framework for handling secure access, user identities and roles. It allows the creation and management of user profiles, which include essential information such as name and password. In the context of the platform, user management serves as the backbone for facilitating effective collaboration between users.

The user management is implemented in its simplest form in the current version. Every user must authenticate before accessing the application (Figure 5.4). To improve the user experience and simplify the initial registration, users are able to use social providers like Google to authenticate. On their first login, users are prompted with a dialog to enter their name. If the name is already known through the social provider, users only have to confirm it. The name can be edited at any point in time (Figure 5.5).

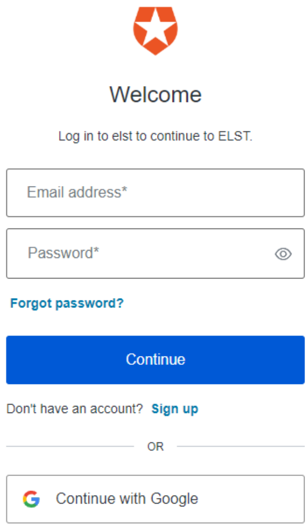

Figure 5.4.: Auth0 Login Screen

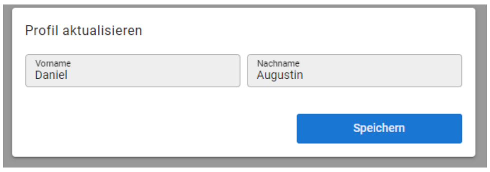

Figure 5.5.: Edit User Profile Dialog

5.3.2. Lesson Planning

The lesson planning feature empowers educators to design and organize their instructional content, providing a structured foundation upon which effective e-learning courses can be built.

The feature is designed in a way that educators do not have to deal with technical details. They are supposed to focus on the structure and content of the lesson as well as their pedagogical goals and content. Figure 5.6 illustrates the model of a lesson plan. The model is based on material and informal exchanges with students from the Pädagogische Hochschule Karlsruhe (Figure A.1).

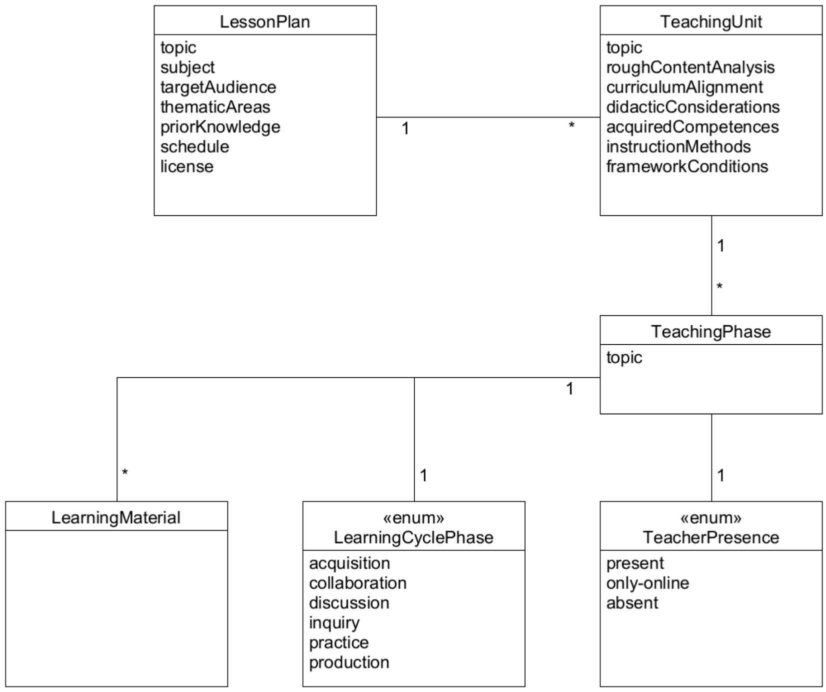

Figure 5.6.: Lesson Plan Model

A lesson is structured around a central topic that it aims to cover comprehensively. This topic is typically associated with one or more subjects and spans various thematic areas. Each lesson is designed with a specific target audience in mind, tailoring the material to their needs and level of understanding. To ensure effective learning, the lesson also assumes certain prior knowledge, building on what the learners are expected to already know. The delivery of the lesson is scheduled to fit within a broader educational framework, ensuring consistency and coherence across the curriculum. Additionally, the availability of the lesson is governed by a license, which outlines the terms under which the material can be accessed and used by other educators.

A lesson can be divided into multiple teaching units. A teaching unit can be thought off as a single classroom session in a traditional education setting. Each teaching unit covers a topic. Educators can provide a rough content analysis to describe the material covered and document their domain knowledge for developers to reference in subsequent development phases. The content is usually derived from a curriculum, and this relationship can be recorded in the curriculum alignment section. Furthermore, educators can document the acquired competencies, didactic considerations, and instructional methods, outlining their pedagogical design. This documentation ensures that the teaching approach is clear and consistent. Lastly, educators can specify the framework conditions for the teaching unit, including available materials, location, and any other constraints that may affect the delivery of the unit.

5.3.3. Course Planning

Once a lesson plan is finalized, it serves as the foundation for the subsequent course planning. The platform facilitates the transition from lesson planning to course planning by allowing educators to convert their lesson plans into course concepts. This includes specifying the technologies that will be used to deliver the course. The integration between lesson planning and course planning ensures that pedagogical goals are aligned with technical implementation. Developers can refer to the detailed lesson plans to understand the educational objectives and content structure, allowing them to create more effective and engaging e-learning courses. The model of the course concept is illustrated in Figure 5.7.

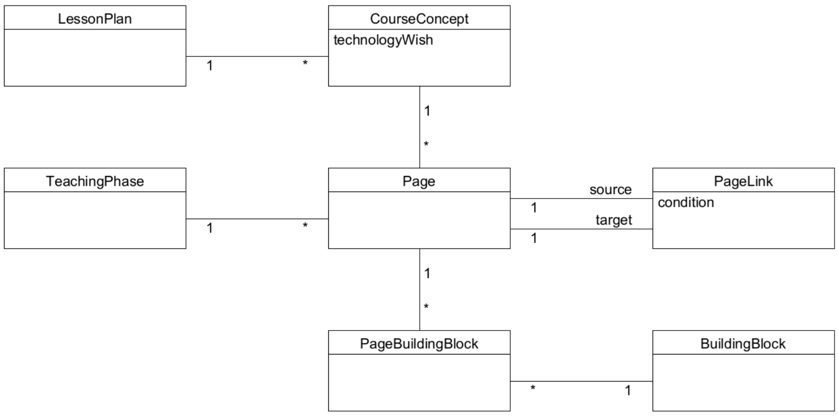

Figure 5.7.: Course Concept Model

A course concept is always based on a previously created lesson plan. Multiple course concepts can be based on the same lesson plan. Educators can suggest a technology when they create the course concept. This technology wish can be further detailed in collaboration with developers. The lesson plan is translated into an e-learning course by structuring it into pages. The structure of the lesson (i.e., teaching phases) gives a rough structure to the pages. Pages can be conditionally linked. For example, pages A, B and C can be linked based on the results of a quiz. If students get 65% on the quiz on page A, they get linked to page B, otherwise page C. Furthermore, the course concept prepares the course implementation by selecting suitable building blocks from the building block repository. Building blocks that are selected for a page are referred to as page building blocks. Page building blocks can be configured based on the exposed properties. This allows educators to adapt the building blocks to the context of the page.

5.3.4. Building Block Development

During the creation of a course concept, suitable building blocks are selected from a central building block repository. This repository contains available building blocks and makes them searchable. The current version of the platform implements a simple version of building blocks. Educators and developers can browse through a list of reusable building blocks and incorporate them into their course concept (Figure 5.9). If no suitable building block exists, a custom building can be requested (Figure 5.10). An initial description serves as a starting point for subsequent refinements. Figure 5.8 illustrates the building block model.

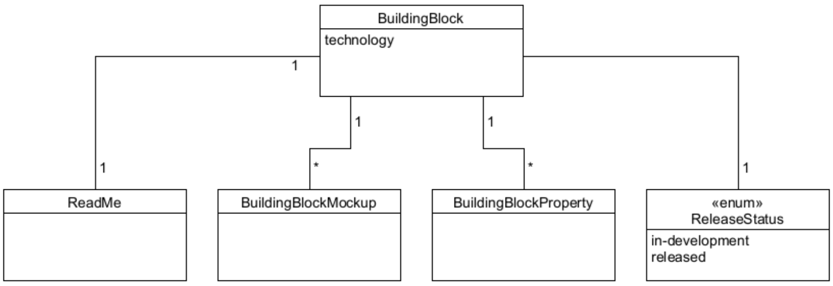

Figure 5.8.: Building Block Model

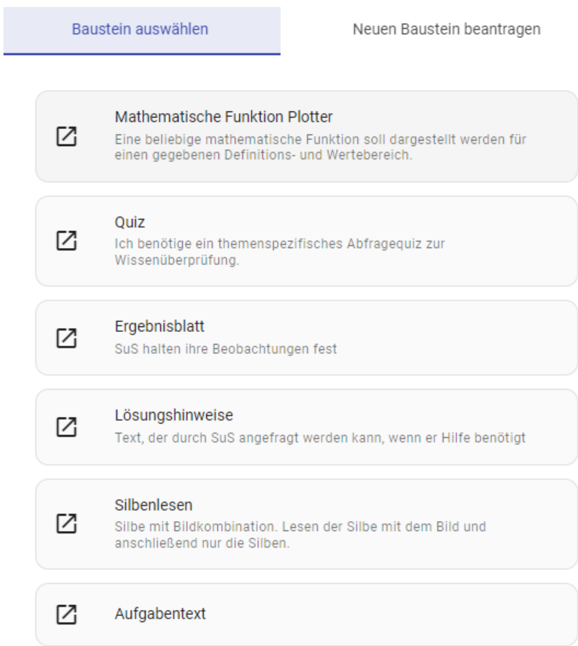

Figure 5.9.: Browse Building Block Repository

A building block is created for a specific technology. This allows developers to assess whether it can be incorporated into a course concept. A readme serves as a user-friendly description of the building block akin to projects hosted on open-source platforms like GitHub [15]. Mockups of the building block allow developers to showcase its visuals. To allow the configuration of

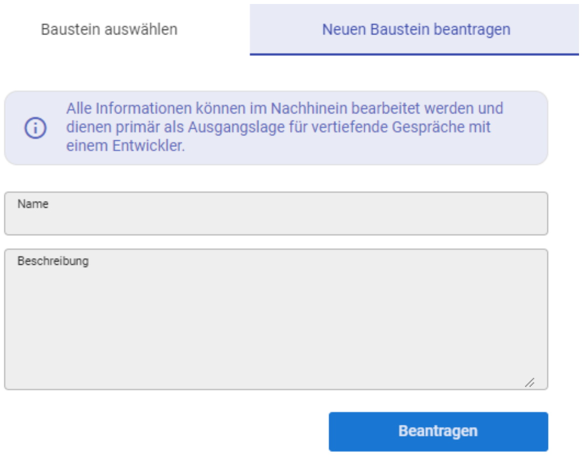

Figure 5.10.: Request Building Block

building blocks to adapt them to the context they will be used in, properties can be specified. A simple "Image" building block might have three properties: image source, dimensions and caption. These properties can be configured after a building block is added to a page inside of a course concept.

A building block can be added to pages as soon as it is requested. However, the configuration of the building block is only enabled for released building blocks. This allows developers to freely experiment with the properties without having to worry about consumers. After the initial development of a building block is completed, its release status changes from in-development to released. The building block can be configured at this point.

5.3.5. Communication

Collaboration is at the heart of the platform and is designed to bring educators and developers together. A key to support collaboration are integrated communication features. A key assumption is that both educators and developers are limited in the time they can spend on the platform (Section 4.1). Therefore, the communication features must support an asynchronous communication style. The current version supports three ways to communicate that require different degrees of involvement and effort.

The simplest form of communication are implementation status indicators for pages (Figure 5.14). Developers can indicate whether they have started working on a page, are waiting for feedback, or have completed implementing a page. This allows educators to quickly scan the progress and decide whether they need to provide more input. However, the status indicators do not carry a lot of information.

Implementation notes are another way to communicate through the platform (Figure 5.15). These notes are suited to provide additional information, requirements or hints. Educators and developers can use these notes to document and share their ideas. Implementation notes can be documented globally for a course concept or on individual pages. Implementation notes can also be provided during the specification of building blocks.

The chat functionality enables direct communication among users, allowing stakeholders to discuss ideas, share updates, and resolve issues. In the current version, this feature is implemented as discussions. Figure 5.11 illustrates a simplified model of this feature. Discussions about a topic can be initiated by any user. Users who want to participate in the discussion can post comments. Initially, the discussion is open. After the discussion ends, it can be resolved to indicate that no further comments are necessary. Users can reference existing building blocks, pages or mockups in the discussion. Figures 5.12 and 5.13 illustrate the user interface of the discussion feature.

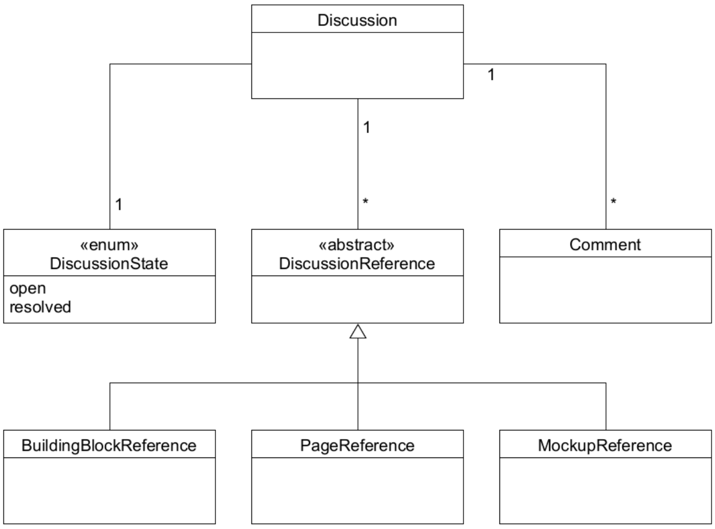

Figure 5.11.: Discussion Model

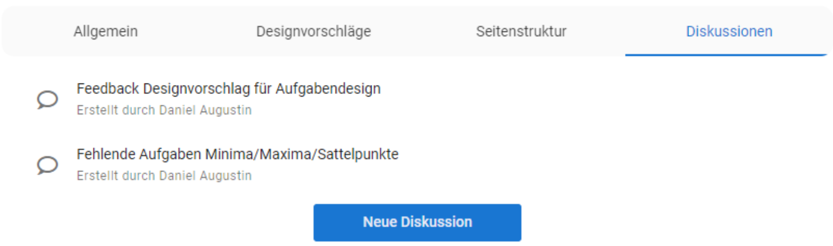

Figure 5.12.: Discussion Explorer

5.4. Architecture

The architecture of ELST follows a client-server model which is illustrated in Figure 5.16. The application logic resides in the backend of the platform. The backend provides a REST API,

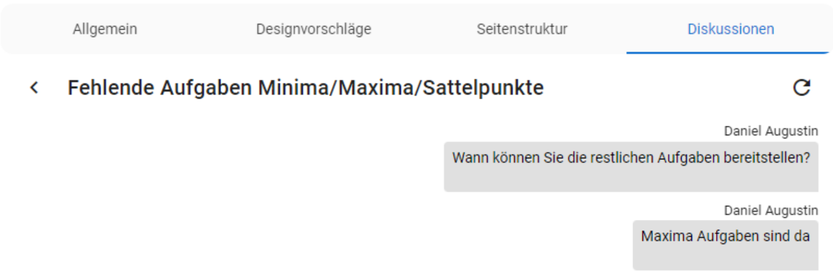

Figure 5.13.: Discussion Chat

which is consumed by frontend clients. The frontend is responsible for displaying the user interface, handling user interactions, and displaying data retrieved from the backend. All application data like lesson plans and course concepts is stored in a SQL database. To simplify the implementation of secure authentication and user management features, an external identity provider is used. It integrates with the frontend and backend through the OpenID connect (OIDC) protocol. The frontend requests an access token from the identity provider and uses the token to authenticate against the backend API. The backend calls the identity provider to validate the access token.

5.4.1. Backend

The backend is structured as a modular monolith [52] to simplify the development and quickly iterate on the features described in Section 5.3. This approach organizes the application into loosely coupled modules with well-defined boundaries and explicit dependencies on other modules. This enhances the maintainability of the application and should ease future development efforts to extend the platform [52]. Figure 5.17 illustrates the modules that were defined for the current version. The modules closely align with the desired features. The lesson planning module encapsulates the functionality to create lesson plans. Similarly, the course planning module enables users to create course concepts based on previously created lesson plans. The building block module provides features to request building blocks and browse the building block repository. Some features include the upload of files which is handled by the content upload module. For example, educators can upload learning materials for each teaching phase or developers can upload mockups for building blocks and pages. The content upload module also allows the download of uploaded files. Communication features are encapsulated inside the communication module.

The backend provides a REST API that can be consumed by clients to use the features of the application (Section 5.3). The API is specified using the OpenAPI standard [44] (Figures A.5, A.6, A.7, A.8). The specification files can be used to generate server and client side code reducing the implementation overhead and ensuring that client and server are compatible [11].

Java 21, the latest long-term support (LTS) version at the time of writing, is used to implement the backend. This ensures stability and access to the latest features and improvements in Java.

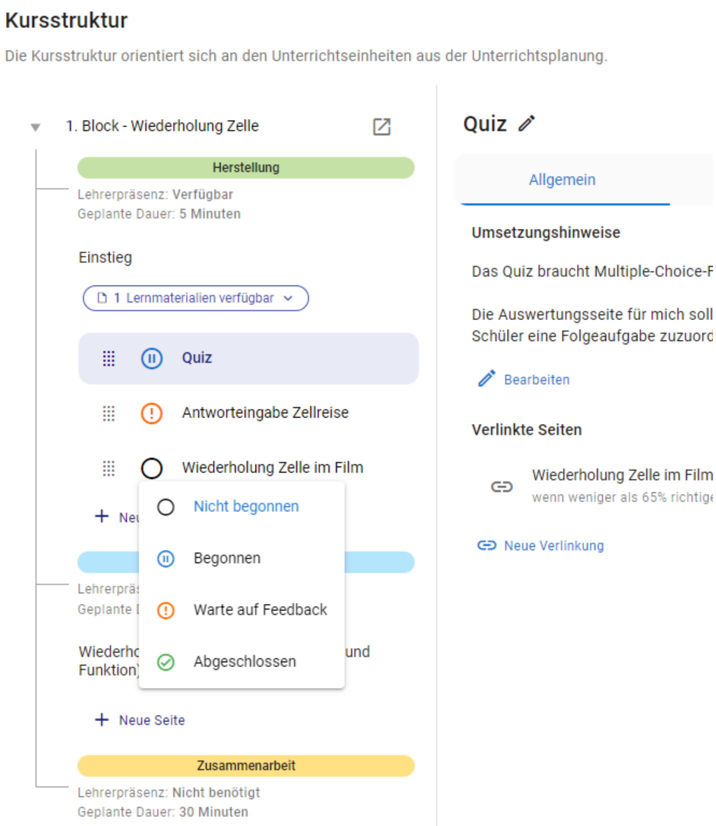

Figure 5.14.: Course Structure with Page Status Indicators

The Spring Boot framework is chosen for its robust set of features, including an embedded web server, REST APIs, security, and data access libraries such as JPA [53]. This choice leverages the widespread use and familiarity of Java and Spring Boot. Particularly among students at the Karlsruhe Institute of Technology (KIT) it can be assumed that they are familiar with Java. This familiarity is expected to facilitate the continued development and maintenance of the platform. Additionally, Spring Boot is compatibile with OpenAPI generators.

The backend does not include very complex logic because the API mainly revolves around simple create, read, update and delete operations. Given this straightforward logic, the primary focus of testing has been on integration tests, which ensure that common use cases function as expected. These integration tests start the entire application, including a database, and assert the responses of the REST API. This method guarantees that all layers of the application work together. This approach aims to balance thorough testing with the limited time constraints present during this thesis. The goal is to prevent regressions that could disrupt critical user interactions, such as communication, lesson planning, course planning, and building block development. By focusing on integration tests, the platform ensures that key features remain functional and reliable without requiring an extensive and time-consuming testing process.

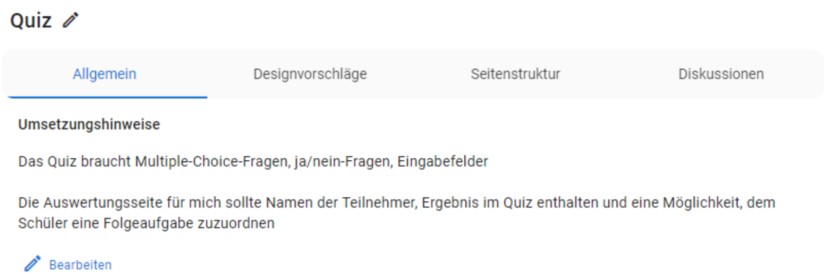

Figure 5.15.: Page Implementation Note

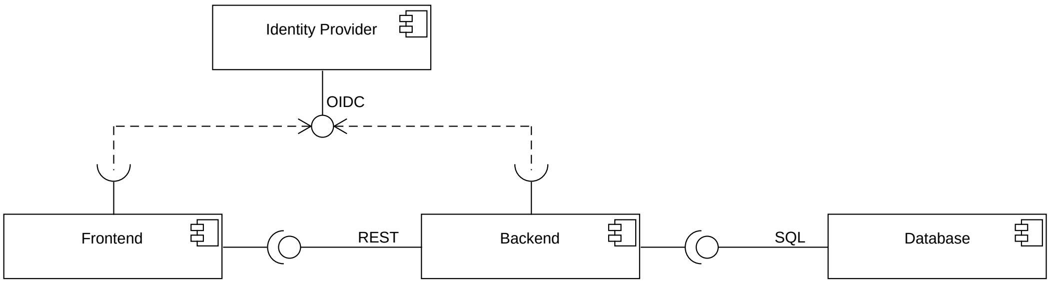

Figure 5.16.: Architecture Overview

5.4.2. Frontend

The frontend of the platform is designed to provide users with an intuitive and user-friendly interface, particularly for non-technical users such as educators. Vue 3 was selected as the frontend framework due to its flexibility and ease of use. Vue allows developers to create web user interfaces using HTML, CSS, and TypeScript, making it approachable for new developers. The decision to use Vue 3 is also informed by its previous successful use within the institute. To enhance the UI further, the Quasar Framework [47] is employed as a user interface component library. Quasar offers a comprehensive set of components that streamline the development process and improve the overall user experience. The author's extensive prior experience with Quasar contributes to this choice, ensuring efficient and effective implementation. Moreover, Quasar supports the creation of progressive web apps (PWAs), which combine the broad reach of web applications with the rich capabilities of platform-specific apps [16]. For example, PWAs can receive push notifications, a feature that could be valuable in future versions of the platform to notify users of relevant updates and events.

5.4.3. Identity Provider

Identity providers (IdPs) offer a range of services to manage user authentication and authorization securely and efficiently. By using an off-the-shelf solution, the development efforts can concentrate on core features rather than dealing with the complexities of identity management.

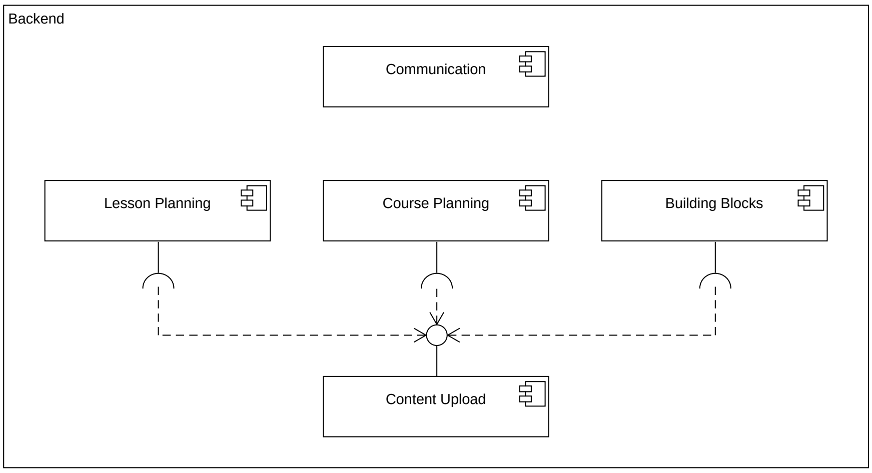

Figure 5.17.: Backend Modules

Using an identity provider significantly enhances security because it eliminates the need to store and manage user passwords, which is a common source of security breaches [13].

Auth0 [5] was chosen as the IdP for this platform due to its comprehensive, secure, and scalable solution for managing user identities, authentication, and authorization. One of the key benefits of using Auth0 is its generous free tier, which supports the initial phases of development and deployment without incurring significant costs. Additionally, Auth0 provides a client SDK that seamlessly integrates into the frontend, simplifying the implementation process [4]. The author's prior experience with Auth0 further supports this decision, ensuring a smoother integration and leveraging existing knowledge.

OIDC is an interoperable authentication protocol based on the OAuth 2.0 framework, simplifying the verification of user identities based on authentication performed by an authorization server. This allows users to sign in with their Google account for example which enhances the user convenience [13]. Since no Auth0-specific features are used, vendor lock-in can be avoided. Auth0 can be easily replaced by a different solution that supports the OIDC protocol.

5.5. Deployment

This section describes how the individual components of the platform are deployed (Figure 5.18).

Auth0 was chosen as the identity provider. The Auth0 service can be accessed through their cloud service offering.

The frontend is realized as a Vue application and uses client-side-rendering. Thus, the frontend is running directly on the client machines. The frontend is delivered as static files that are downloaded when a user opens the application in their browser.

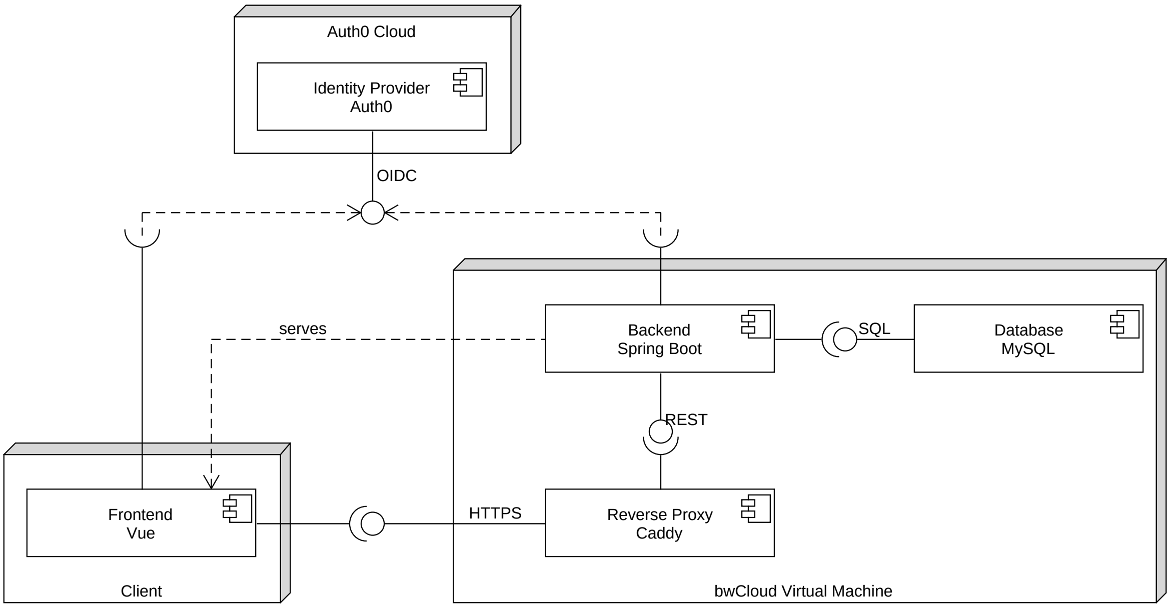

Figure 5.18.: Platform Deployment

The backend and database were deployed on a virtual machine hosted by the bwCloud [9] which offers free virtual machines to students. To establish a secure HTTPS conncection between the frontend and backend, a reverse proxy is required to terminate the TLS connection. Caddy was selected as a reverse proxy because it does not require a complicated setup and is able to automatically provision TLS certificates [50]. The backend is realized as a Spring Boot application which has an integrated web server. Therefore, it can be used to serve the frontend which simplifies the deployment.

The deployment is handled with the help of a docker compose script (Listing A.4). This script starts three docker containers: reverse proxy (Caddy), backend (Spring Boot) and database (MySQL). All containers are placed inside the same docker network allowing the containers to interact with each other. The database is not publicly exposed and can only be reached through the secured backend. The images for caddy and MySQL are downloaded from the docker registry. The caddy configuration is provided in a separate file (Listing A.2). The backend image is built using a provided Dockerfile (Listing A.2). This Dockerfile first builds the frontend. The resulting files are then copied to the static files folder of the Spring Boot application. Afterwards, the Spring Boot application is tested and packaged into a JAR file which is then used as the entrypoint of the backend image. This solution ensures that the entire application, including both frontend and backend, is packaged and deployed as a cohesive unit. Currently, the deployment process is performed manually. However, it can be easily automated using a deployment pipeline.

6. Evaluation

This chapter presents the evaluation of the collaborative development approach (Chapter 4) and the collaborative e-learning development platform (Chapter 5).

Since the development approach and platform are completely new and only a subset of features are implemented in the current version, the evaluation of the approach and platform is challenging. Common evaluation methods found in the related work (Chapter 3) are case studies. During these studies, the proposed development approach is used to either create learning objects or courses. This is considered to be an ideal approach for evaluating a devleopment approach [20].

However, this evaluation method is not suitable for the current state of implementation because many important features like course implementation, course deployment and project management are missing. Thus, the focus of the evaluation in this thesis is shifted towards examining foundational assumptions and identifying limitations of the proposed approach. The goal is to identify areas of strength and weakness in order to provide guidance for future development efforts.

The employed methods are further described in Section 6.1. The results of the evaluation are described in Section 6.2.

6.1. Methodology

To identify areas of strength and weakness in order to provide guidance for future development efforts, a combination of methods found in the related work are employed. First, the comparison framework proposed by Montilva et al. [36] is applied to the collaborative development approach described in Chapter 4. The comparison framework defines attributes that development approaches in the e-learning domain may exhibit. By applying the comparison framework, possible shortcomings and areas of improvement may be identified. The framework provides a checklist that could uncover general requirements for the proposed development approach (RQ-1). Additionally, a case study with interviews and surveys is conducted to gather insights from the perspective of educators. The goal is to elicit requirements for a collaborative development approach (RQ-1) and to identify key features of the platform (RQ-2). Interviews allow educators to freely express their needs and discuss them. Only educators will be considered for the case study. This decision is due to limited time constraints and the fact that the implemented features of the platform are mainly geared towards educators.

6.1.1. Development Approach Comparison Framework

Montilva et al. [36] designed a comprehensive comparison framework that enables the identification of the main advantages and shortcomings of their method relative to other well-documented methodologies in the literature. This framework was further utilized by Hadjerrouit in 2010 [20]. The comparison framework is composed of four views: domain view, usage view, product view, and process view. Each view is characterized by a set of facets, which in turn are composed of attributes that describe different features of the view. The facets, attributes, and their values have been adapted to characterize methods devoted to building instructional websites.

Domain View pertains to the application domains of the methods being compared. It is described by two facets: the scope facet and the instructional facet. The scope facet allows the evaluator to assess the methods according to their coverage of technical, instructional, and management features, as well as their application area. The instructional facet focuses on the educational features of the websites developed by the methods. It includes considerations of the educational level, instructional modality, type of course to be developed, and the dependence of the methods on specific instructional approaches. This facet is vital for understanding how well the method supports educational objectives and integrates pedagogical principles.

Usage View addresses the applicability and usability properties of the method. These properties are captured by three facets: applicability, usability, and user involvement. The applicability facet evaluates the phases of the website lifecycle covered by the methods and their primary uses. The usability facet describes the usage characteristics of the methods, ensuring that they are user-friendly and accessible. The user involvement facet specifies the expected types of users and their participation in the course site development process, highlighting the importance of including educators and developers in meaningful ways.

Product View evaluates the product models used by the methods. It is described by two facets: the product representation facet and the conceptual description facet. The product representation facet characterizes the product model availability, its orientation, the notation used to describe it, and the set of perspectives or points of view used to typify the product. Having an explicit course site representation is critical for understanding the result to be obtained by using the method, as it provides the development team with a comprehensive vision of the product structure and functionality. Moreover, the product model determines the characteristics of the process model of a method, making it a key component of the evaluation. The conceptual description facet is concerned with the types of concepts used by the models that are explicitly stated in the method documentation. This facet ensures that the product models are well-defined and grounded in clear conceptual frameworks, which is essential for consistency and clarity in the development process.

Process View contains the set of facets and attributes that delineate the process of developing a course site. This view evaluates the process models used by the methods in terms of five facets that consider the orientation and approach, the characteristics, and the types of processes covered by the models. By examining these facets, evaluators can gain a detailed understanding of how the methods structure the development process, including the steps involved, the roles of different participants, and the overall workflow.

6.1.2. Case Study

The goal of the case study is to gather insights into the comprehensibility of the development approach, the ability to consider pedagogical design decisions, and the usability of the platform from the perspective of educators. The scope of the interview and survey must be carefully planned to fit within the limited time that educators can allocate, given their already busy schedules. Figure 6.1 illustrates the steps of the case study.

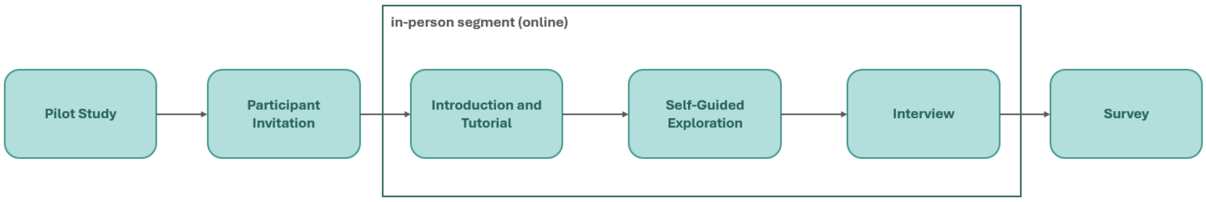

Figure 6.1.: Case Study Steps

Before the case study, a pilot study was conducted to practice the procedure and prevent potential participant confusion. A student teacher working at the department was selected for the pilot study. The participant was shown the invitation letter and went through all the steps of the in-person segment and completed the survey. The feedback from the field study was incorporated into the final development iteration of the platform.

Afterwards, participants for the case study are invited. The only requirement is that the participants are educators or studying to become educators. The goal is to include participants from different backgrounds. For each participant, an online meeting appointment is scheduled for the in-person segment. The in-person segment includes three steps.

In the beginning of the in-person segment, the participants are introduced to the topic. It is assumed that many participants are not fully aware of the possibilities e-learning provides. The participants are sensitized by mentioning possible advanced features like customized feedback, integrated learning analytics and different content based on student performance. Afterwards, the participants are introduced to the platform by showing an example course. It is explicitly mentioned that the content of the lesson plan and course are irrelevant to avoid educators focusing on details and to reduce the stress from being observed. After the tutorial, participants are informed that the following steps are recorded. The recording is only saved locally and will be transcribed. Afterwards, the recordings are deleted to comply with privacy laws. The participants will be anonymized and only identified through a unique identifier. The identifier is assigned to each participant prior to the in-person segment and is used to correlate the interviews and surveys. For the self-guided exploration of the platform, participants are invited to plan their own lesson and derive a course. During this phase, participants are encouraged to think aloud [10]. The researcher guides the participants and ensures that the agreed time is not overdrawn. The in-person segment is scoped to 1.5 hours. If participants struggle with finding ideas, a set of predefined scenarios is provided. These scenarios are designed to be simple examples that highlight specific features or require collaboration. The following list briefly describes the prepared scenarios:

Scenario 1 Students should be shown different learning materials based on their results from an interim test.

Scenario 2 Hints should be displayed based on the time spent.

Scenario 3 Developers have created an initial design proposal and need feedback.

Scenario 4 Developers do not understand the written description and request that the educator uploads a sketch.

Scenario 5 The educator wants to receive an email with the students' results upon course completion.

Scenario 6 An existing learning resource created by the educator should be integrated.

Scenario 7 Students should be shown hints and additional learning materials based on the types of mistakes they make when answering questions.

Collaboration features are showcased by simulating a developer. The developer is simulated by the researcher who has a background in software engineering. Therefore, only simple collaboration examples can be showcased. This is not ideal because the collaboration is designed to be asynchronous and span multiple days. However, this setup significantly simplifies the case study and ensures that educators do not have to spend too much time.

For the interview, an interview guide [1] was created to comprehensively gather insights from participants. The original interview guide is accessible in the KIT thesis repository (Chapter 8). It includes eight main questions, each targeting specific areas of interest. Most questions have supporting questions to help participants if they do not know how to answer. The first question focuses on identifying functions that should be prioritized in future developments. The second question aims to evaluate the comprehensibility of the development process. The third question assesses the meaningfulness of the development process, particularly in terms of transferring a lesson plan into an e-learning course. The fourth question explores various collaboration points within the platform. Questions five, six, and seven evaluate the implemented communication methods and collaboration points in more detail. Finally, the eighth question seeks to identify features that participants find particularly valuable.

Interview Guide (translated)

1. Q1: What qualities make an e-course valuable for you?
    a) Would you implement your in-person teaching in the same way as in an e-learning course?
    b) What are the reasons why you have not yet used an e-learning course?

   c) Are there functions that are only made possible through e-learning courses?
   d) Are there functions that, in your opinion, cannot be represented through e-learning courses?

2. Q2: How did the structure of the application help you in creating the e-learning course?
   a) What differences did you notice compared to other tools?
   b) Does the structure of the application meet your expectations? How would you structure the application?
3. Q3: How well could your lesson plan be implemented into an e-learning course?
   a) What difficulties did you encounter in the process?
4. Q4: In which areas of course development do you wish for support from developers? Do you find the separation of activities (lesson planning, course planning, course implementation) useful?
   a) Was the separation clear and understandable?
   b) Did you perceive the separation as useful?
5. Q5: Which communication methods (status display, notes, chat) do you prefer when collaborating with developers? Why?
6. Q6: Were there obstacles in communicating with developers?
7. Q7: Which collaboration features did you find particularly valuable, and how could collaboration be further enhanced?
   a) Which features should be further developed?
8. Q8: Which features of the application did you particularly appreciate or find problematic, and why?
   a) Which features should be further developed?

Lastly, the participants were given a link to the survey. The researcher is not present during the survey to avoid participants feeling pressured. The survey contains general questions about the background of the participant. Furthermore, questions about usability and technology acceptance were asked. The Technology Acceptance Model (TAM) and the System Usability Scale (SUS) are used. Both had to be translated to German because participants likely speak German.

In empirical research on technology acceptance, the TAM is predominant. For this study, the TAM questionnaire published by Venkatesh and Davis [56] was used. The German translation of the questionnaire was conducted by Olbrecht [43]. The answers to the questions about the TAM were recorded using a five-point Likert scale: "1: strongly agree" to "5: strongly disagree".

The usability of the platform is assessed using the SUS proposed by John Brooke [8]. It contains ten questions about the usability of a system. The SUS is "a useful tool to understand the problems of users facing while they are using the system" [25]. SUS stands out with its wide range of usage areas, simplicity, and quickness of use for both researchers and participants. SUS provides a general overview of the usability of a product with the help of its score calculation. Although the score does not give an absolute judgement on the usability of a product. To address this, a five-point Likert scale is added as the eleventh question. The question is: "Overall, I would rate the user-friendliness of this product as:" and the answer to this question ranges from "1: Worst imaginable" to "7: Best imaginable". The adjective rating scale could help to find an absolute judgment from the SUS questionnaire [25]. The German translation of the questions was taken from [14].

The questionnaire was created as a Google Form, which has previously been used in the department. It allows for easy distribution of results while maintaining participant anonymity, as Google can be configured to not store email addresses.

6.1.3. Limitations

There are several significant limitations to consider. Firstly, the chosen methods do not provide definitive answers regarding the overall efficacy of the proposed collaborative development process in successfully creating e-learning courses. The evaluation does not test whether the approach effectively supports collaboration between educators and developers, a critical component of the approach. Moreover, excluding developers from the evaluation omits a key stakeholder, whose insights are vital for a holistic assessment.

The scope of the case study is also restricted due to time constraints. The participants, being volunteers, are likely to have a predisposed positive attitude towards e-learning. This could skew the results, as these participants may not represent the average educator's experience and perception. Additionally, the presence of the researcher during the in-person segment of the case study might influence the participants' behavior and feedback. This could lead to less authentic responses. Furthermore, the timing of the survey, conducted after the in-person segment, allows participants to complete it later. This delay might lead to altered results since participants might not accurately remember their experiences and the platform's features. The availability of educators is likely very limited due to educators' busy schedules and the timing of the study during the holiday season. This presents a critical limitation, as it restricts the ability to generalize the findings. A larger sample size would be necessary to draw more robust and reliable conclusions. Finally, participants' varying levels of familiarity with similar technologies could significantly affect their ease of use and feedback. More tech-savvy educators might find the platform more intuitive and user-friendly compared to those less familiar with technology, leading to a disparity in the feedback received and an inaccurate assessment of the platform's usability across a diverse user base.

The comparison framework by Montilva et al. [36] is utilized not for direct comparison with other development approaches but to identify potential shortcomings and areas for future improvement. The comparison framework is not tailored to assess the collaborative aspects of the approach, thus limiting its effectiveness in evaluating one of the core premises of the proposed method.

These limitations indicate that further studies are necessary. At the time of this thesis, the development approach and platform are still in their early stages. Consequently, the focus is on identifying limitations and possibilities for future improvement. Despite the limited sample size and scope, the case study can still gather valuable feedback. The presence of the researcher during the in-person segment is essential to address potential bugs and explain features directly to the participants. Although the exclusion of developers is not ideal, future versions of the platform will include features relevant to developers, making their feedback more valuable once these features are realized. Participants will be encouraged to complete the survey immediately after the in-person segment to reduce the impact of altered memories. Any potential differences due to familiarity with similar technologies are mitigated by selecting participants with diverse backgrounds.

6.2. Results

This chapter presents the results of the evaluation which was performed between 14.06.2024 and 21.07.2024. The interview guide, interview transcripts and survey results are available in the KIT thesis repository (Chapter 8). The section begins with a brief description of the involved participants. Following this, findings from the interviews are presented, showcasing the qualitative feedback from educators regarding their experiences with the platform. Afterwards, the survey results are presented, providing quantitative data on usability and technology acceptance. Finally, the chapter concludes with an analysis based on the comparison framework, which offers a systematic evaluation of the development approach across various dimensions.

6.2.1. Participants

Two invitation rounds were necessary to find participants for the case study. In the first invitation round, twelve e-mails were sent to educators with ties to the department. However, only one educator responded but ultimately did not participate. In response, the search for educators was expanded by posting notices in schools. Additionally, teachers from the extended social circle (friends' teachers) were directly contacted via e-mail. In total, notices were posted in three different schools and five additional e-mails were sent. The emails included background information, details about the time commitment, and contact information for scheduling. Participants were given the option to choose between an online meeting or an in-person appointment. No specific time restrictions were mentioned for scheduling, allowing teachers as much flexibility as possible.

Eventually, four educators agreed to participate (Table 6.2.1). The participants are between 30 and 60 years old and have an even split between male and female. They have diverse backgrounds with varying degrees of prior experience in e-learning. The taught subjects include german, mathematics, geography, history, music and religion. Two educators are currently employed at a german high school (P1, P2). Both have previously used Moodle and MS Teams at their school. The next participant (P3) recently concluded work abroad where he helped other educators to use a pre-made e-learning course. The last participant (P4) used to

Participant	Gender	Workplace	Subjects
P1	female	High School	Biology
P2	female	High School	Math, Music
P3	male	High School	History, Geography, Russian
P4	male	School in Prison	German, History, Geography, Religion

Table 6.1.: Case Study and Survey Participants

work at a German high school but is currently employed as an educator in a german prison. The prison uses the Elis e-learning platform [22] developed by the TU Berlin which provides an extensive collection of digital learning materials which can be used by educators to support their classroom teaching. The participant is involved in promoting and improving e-learning in the prison.

6.2.2. Case Study Findings

The interviews conducted with educators regarding their experiences with the collaborative e-learning platform revealed several insights. The original interviews were conducted in German and have been translated into Englisch for the purpose of this report. Unfortunately, the recordings of two participants were corrupted making them unusable for the evaluation. Thus, only the interviews with P1 and P4 are used in the following results.

All interviewed educators had prior experience with e-learning tools. They emphasized the value of personalized feedback and adaptive learning paths in enhancing student engagement and learning outcomes. Educators found self-check activities and quizzes particularly useful for maintaining student interest. E-learning platforms' ability to provide immediate and automated feedback is highly valued, as it supports motivation and self-paced learning. P1 explained that personalized feedback is hardly possible without e-learning: "You can't possibly manage that in a classroom as a teacher with 30 children" (I-P1-1). This real-time feedback is seen as crucial for maintaining student motivation and promoting continuous learning. Moreover, educators preferred a blend of traditional and e-learning methods rather than relying solely on e-learning, as this hybrid approach allows for greater flexibility and adaptability in teaching. The ability to use and adapt existing apps and tools was especially advantageous for diverse classrooms, offering opportunities for differentiated instruction that cater to varying student needs and levels. P4 highlighted the ability of supporting multiple languages "because I have so many different nationalities" (I-P4-1).

P1 previously used e-learning tools, such as Moodle and Kahoot, as well as self-check tools. However, P1 noted that Moodle's design had become outdated and less appealing to students, who found it less engaging compared to modern tools they use in their personal lives. Additionally, frustration was expressed with the process of copying and reusing materials in Moodle, as well as the challenges in integrating feedback effectively. Technical limitations within schools and varying levels of technical proficiency among educators were cited as factors that hinder the effective use of e-learning tools.

The educators overall expressed interest in the idea of a collaborative e-learning development platform: "I think that is a fantastic idea you have [...] Bringing [educators and developers] together is a great thing, and I would definitely like to keep exploring it further" (I-P4-2). Educators appreciated starting with a lesson plan because it let them focus on their pedagogical goals and not get hung up on details. P1 even went so far as wanting to "create my entire yearly or semi-yearly plan, count all the available hours, and distribute the topics to ensure I reach the end goal" (I-P1-2). However, the detailed lesson plans that can be created with the platform were criticized. Detailed lesson planning emphasized during teacher training often diminishes with years of experience. Instead, they appreciate the flexibility to adapt lessons based on their accumulated experience and previous materials. With increased teaching experience, the need for precise and detailed planning tends to decrease, allowing for more spontaneous and responsive teaching methods. Both P1 and P4 remarked that after 20 years of teaching, detailed phase-by-phase planning becomes less necessary, as much of the process is internalized and familiar themes can be adjusted easily. Educators expressed a preference for a more simplified, less segmented approach to lesson planning. One teacher mentioned, "I would make it a bit more stripped down, less in these teaching steps" (I-P4-3) indicating a need for a more streamlined process.

The collaboration between educators and developers emerged as a crucial area for improvement. There is a clear need for precise communication to ensure that the tools developed meet educational needs effectively. Educators desired efficient communication channels and notifications about updates or changes because they do not want to spend too much time on the platform. A significant concern was the inefficiency in communication between educators and developers because it is hard to precisely formulate text-based requirements so they are understood by everyone. P1 pointed out that while having technical assistance can reduce certain obstacles, it introduces new challenges, particularly when precise and detailed descriptions are required to convey what is needed. This process often feels unnatural, as it transforms the educator from a creator to a client who must clearly outline their requirements. P1 noted that this shift in roles — where the teacher becomes an 'orderer' rather than a content creator can be somewhat uncomfortable. They highlighted that while many educators are accustomed to searching for and adapting existing resources from the internet or digital publishers, having a developer to create custom materials introduces a new dynamic. The ability to fine-tune and adjust materials themselves, rather than depending entirely on external developers, would be a significant advantage. This self-sufficiency is seen as a preferable alternative, allowing educators to maintain greater control over their instructional materials. P4 remarked, "You as a developer wouldn't be satisfied with the information you received from me. And I wouldn't be satisfied with you because you didn't deliver it in the time I needed" (I-P4-4). Thus, P4 felt more comfortable creating and adjusting their materials independently, interacting with developers only when necessary to reduce the potential of communication mistakes.

P4 discussed their experience with the Elis platform that provides a collection of apps as a resource repository, functioning much like a learning search engine. P4 appreciates the freedom to select and integrate tools according to their specific teaching needs, rather than relying heavily on ongoing developer support. They highlighted the advantage of these pre-made apps, noting that they often come with sophisticated features such as gamification and immediate feedback mechanisms. These features not only engage students through reward systems but also streamline the assessment process by providing automated grading. The educator remarked that this level of functionality is something they could not achieve within the same timeframe if they had to develop the tools themselves or with developers. The building block features provided by the application in the current version are too limited to provide the same level of freedom to educators. P1 also expressed the need to use building blocks on their own: "If I know I want to use a quiz in my lesson, I can simply request that specific quiz" (I-P1-3). P1 also remarked that "when I think back to Moodle, I remember building my quizzes myself back then" (I-P1-4). Reusability of lesson plans and building blocks was highlighted as a key requirement of effective e-learning platforms.

Feedback regarding the usability of the collaborative e-learning platform identified several areas needing improvement. Educators noted that clickable elements needed stronger contrast, especially when hovered over, to enhance usability. There was also confusion regarding the application and social forms within the learning cycle phases, with one educator commenting on the mix between classroom activities and social interaction, indicating a need for clearer differentiation and structure. The platform's user interface and instructions appeared insufficient, leading to uncertainty and difficulty in use. One teacher expressed this by saying, "I would have needed to write it much more precisely into my learning component, right?" (I-P1-5) and another added, "I had a hard time at the beginning, I admit, because the structure is quite far removed from my usual lesson preparation" (I-P4-5). This shows the necessity for clear, detailed instructions and an intuitive user interface. Uncertainty was a recurring theme, with one teacher mistaking developer notes for student instructions, pointing to the need for better guidance and interface clarity. There was also confusion about a feature to link between course pages which was mistaken as linking external content within courses. Although this was seen as a desirable feature for future versions. Additionally, some educators found it unclear where to place developer notes, requirements and questions within the platform. Educators also wanted to know how students will access the finished course and where students results are accessible.

Overall, the feedback suggests that while the platform has potential, significant improvements in usability, clarity, and interface design are essential for better adoption and effectiveness in educational settings.

6.2.3. Survey Findings

This section presents the results of the survey. The survey included general questions to assess the prior knowledge of the participants as well as questions from the Technology Acceptance Model (TAM) and System Usability Scale (SUS).

Participants were asked to rate how often they use digital aids during their lessons on a scale from 1 to 5. The average of 3.75 indicates that the participants seem to favor the inclusion of digital aids. Only 50% (P1, P4) indicated that they have prior experience in creating e-learning

Question	A-1,	A-2,	A-3,	A-4,	A-5	Average
SUS-1: I think that I would like to use this system frequently.			P1, P3, P4	P2		3.25
SUS-2: I found the system unnecessarily complex.		P2, P3,		P1, P4		3.00
SUS-3: I thought the system was easy to use.		P4	P3	P1, P2		3.25
SUS-4: I think that I would need the support of a technical person to be able to use this system.		P1, P3, P4		P2		2.5
SUS-5: I found the various functions in this system were well integrated.		P4	P1, P3	P2		3.00
SUS-6: I thought there was too much inconsistency in this system.		P3	P2, P4	P1		3.00
SUS-7: I would imagine that most people would learn to use this system very quickly.		P4	P1, P2	P3		3.00
SUS-8: I found the system very cumbersome to use.		P1, P2, P3		P4		2.50
SUS-9: I felt very confident using the system.		P4	P1, P3		P2	3.25
SUS-10: I needed to learn a lot of things before I could get going with this system.	P1, P2	P3	P4			1.75

Table 6.2.: System Usability Scale Results

courses. P1 used Moodle but criticized that students did not find the design appealing. Additionally, the reuse of coures and the evaluation of student results was reported as being difficult. P4 hat prior experience using the elis platform as well as Anton, Bettermarks, Schlaukopf, Seterra, Topeuropa and many more. P4 did not report any problems. The participants indicated that they are interested in new applications to create e-learning courses with an average rating of 4.00 on a scale from 1 to 5.

The results of the SUS are shown in Table 6.2. The table includes the actual answers of each participant. The columns A-1 to A-5 represent the possible answers. If a participant P choose answer option 3, P is placed in column A-3. The average answer is calculated for each question. Most of the results are around 3.00 with slight trends towards the best result. Only SUS-10 had a clear indication that users felt they did not need to learn a lot before using the application. However, there seem to be different opinions when it comes to the perceived complexity and confidence using the system. Participants rated the overall user friendliness with an average of 3 which reflects the overall trend.

Participant	Usability Score (0 - 100)
P1	57.5
P2	70
P3	65
P4	37.5

Table 6.3.: System Usability Scale Scores

Question	A-1	A-2	A-3	A-4	A-5	Average
TAM-1: Assuming I have access to the system, I intend to use it.			P1, P3, P4	P2		3.25
TAM-2: Using the system improves my performance in my job.		P1, P4	P3	P2		2.75
TAM-3: Using the system in my job increases my productivity.		P4	P1, P3	P2		3.00
TAM-4: Using the system enhances my effectiveness in my job.		P1, P4	P3	P2		2.75
TAM-5: I find the system to be useful in my job.		P1	P3, P4	P2		3.00
TAM-6: My interaction with the system is clear and understandable.			P1, P3, P4		P2	3.00
TAM-7: Interaction with the system does not require a lot of my mental effort.		P4		P1, P3	P2	3.50
TAM-8: I find the system to be easy to use.		P4		P1, P2, P3		3.75
TAM-9: I find it easy to get the system to do what I want it to do.			P1, P3, P4	P2		3.50

Table 6.4.: Technology Acceptance Model Results

Table 6.3 shows the usability score of each participant. The score is calculated by assigning points to each question. The questions alternate between positive and negative statements with a scale from 1 to 5. The points for positive statements are the user selection minus one. The points for negative statements are 5 minus the user selection. The assigned points are added and finally multiplied with 2.5. This results in a score that ranges between 0 and 100.

The results of the questions related to the TAM are shown in Table 6.4. The table uses the same structure as Table 6.2. The average answers are very close to 3.00. Only question 7, 8 and 9 score slightly higher than the average. These questions are related to usability which is in line with the results of the SUS.

6.2.4. Development Approach Comparison Framework

This section applies the comparison framework proposed by Montilva et al. [36] to the development approach described in Chapter 4. The goal is to identify areas of improvement based on the views and facets defined by the comparison framework.

Domain View

The development approach is designed to be applied in educational settings, encompassing both instructional and technical activities. Instructional activities include lesson planning and course planning. Technical aspects are covered by the building block development and course implementation phases. The scope of the development approach currently does not include management activities. However, project management features are planned for future iterations of the collaborative e-learning development platform. Therefore, the development approach should be extended to clearly define how management activities integrate with the other phases.

Usage View

The development approach covers all phases of the e-learning lifecycle, including analysis, design, development, evaluation, and maintenance. However, the maintenance phase is only implicitly included by promoting continuous improvement of courses. The usability facet of the comparison framework reveals three areas of improvement: visibility, use of standards, flexibility and efficience of use.

The visibility attribute specifies if it is always clear what to do and how to do it. The proposed development approach does not provide strict step-by-step instructions. It relies on educators and developers to collaboratively drive the development process offering a flexible framework and set of activities that can be freely chosen. For instance, during lesson planning, educators might identify a need for a new building block, prompting a shift to the development phase before returning to planning. The phases can also be performed simultaneously due to asynchronous communication delays. Additionally, the approach provides no specific guidance on performing activities, allowing users to determine the complexity they require. The developed platform offers an opinionated view of the process which defines structures for lesson plans and course concepts but leaves their usage open to user discretion.

The development approach does not prescribe the use of any standard techniques or notations. This is inline with giving users the freedom to choose their desired techniques and notations. However, users might benefit from examples and predefined options to choose from. This would simplify the reuse of lesson plans, course concepts and implementations because users would know what to expect. The developed platform already prescribes a predefined structure for lesson plans based on materials from the Pädagogische Hochschule Karlsruhe (Figure A.1), which can be seen as a standard technique, though details may vary at other institutions.

With regard to flexibility and efficiency of use, the proposed development could benefit from improved documentation including detailed examples and guidelines. Currently, the phases and their relationships to other phases are described. Users of the approach might need more guidance on using the approach in practice. The developed platform is an attempt to provide guidance but it lacks tutorials at the moment.

Product View

The product view evaluates the product models used by the methods. A product model defines the properties commonly attributed to the products developed from different perspectives (instructional, structural, functional, technological). Having a product model available is essential for understanding the outcomes that can be achieved using the method. The proposed development approach does not explicitly define a product model because it is meant to bring educators and developers together to create e-learning courses that fit their requirements without prescribing a structure. However, a product model could help users get started with the approach even if they need to adapt it in the future.

Process View

The process view highlights the need to improve the understandability and guidelines of the approach. The proposed development approach only defines the phases and their relationships. However, it does not provide a well-defined process structure which can be used to get started. Additionally, no guidelines are provided. The developed platform helps by guiding users through the process with its user interface.

The development approach also has potential for improvement in regard to the management process facet. It does not include any project planning and control activities like project plan definition, resource estimation and progress evaluation and monitoring. Quality assurance is also not considered.

Many attributes of the development processes facet are already present in the proposed development approach. However, architectural design and testing activities are not explicitly mentioned in the development approach.

7. Discussion

It is commonly accepted that pedagogy must play a central role in e-learning to create efficient learning experiences (Chapter 3). E-learning, understood as pedagogy empowered by digital technology, has seen the development of numerous approaches over the years (Chapter 3). These approaches vary in focus but consistently emphasize the importance of collaboration among educators, developers, instructional designers, and content creators. Despite this recurring theme, there has been a lack of explicit descriptions on how these roles work together effectively. Some methodologies rely heavily on technical artifacts for communication, which can alienate non-technical stakeholders, such as educators.

This thesis explored the requirements for effective collaboration between stakeholders that ensures the alignment of pedagogical goals and software development processes. The primary goal was to envision a new e-learning development approach that builds on previously proposed methods and explicitly describes how collaboration among stakeholders should function. This approach aims to ensure that stakeholders feel comfortable within their areas of expertise without needing to acquire new, unrelated skills. To narrow the scope of this thesis, only the collaboration between educators and developers was examined. The research sought to understand how the proposed development process meets the individual requirements of educators and developers. Additionally, it investigated the essential features and functionalities that both educators and developers find most valuable in a tool designed to support collaborative e-learning development.

This study is significant as it addresses a gap in the literature by providing a clear framework for collaboration between educators and developers in e-learning development. In addition to proposing this collaborative approach, a platform was envisioned to support its practical implementation. This platform operationalizes the approach, offering guidance for all stakeholders involved. It's goal is to provide an intuitive user interface that is easily accessible, particularly benefiting non-technical users. By aligning pedagogical goals with software development processes, and supported by the platform, the proposed approach has the potential to significantly enhance the effectiveness of e-learning development.

Due to the scope of the envisioned platform, only a subset of its features was implemented during this thesis. Consequently, the approach and platform could not be fully utilized by educators and developers to collaboratively create an e-learning course. The evaluation focused on examining foundational assumptions and identifying limitations. Educators were asked to use the platform and provided their feedback through interviews and surveys. Additionally, a comparison framework proposed by Montilva et al. [36] was employed to identify areas for improvement in the proposed development approach.

The following discusses the results of the evaluation, highlighting the strengths and weaknesses of the proposed development approach and platform. Furthermore, it identifies implications for future development efforts.

Educators overall expressed interest in the idea of a collaborative e-learning development platform, recognizing its potential to enhance the development and delivery of e-learning courses. They identified key features they find valuable (RQ-2) but also pointed out several areas of improvement.

Blended learning is a popular approach to incorporate e-learning into the classroom with all educators. There is a difference between e-learning courses that cover a topic from start to end and simple learning apps that can be selected. Only 50% of educators had prior experience with e-learning platforms like Moodle or Elis. Educators without prior experience had higher usability scores than educators who already created e-learning offerings. P4 is used to the simplicity of picking and choosing learning apps that are freely available on the internet. P4 did not value the focus on creating complete e-learning courses which is reflected in the very low usability score 37,5 out of 100. The platform could benefit from supporting both usage models. If educators are only searching for a specific learning material, they could access the building block repository without having to create lesson plans and course concepts. This could significantly improve the simplicity of the platform and acknowledges real world use cases.

Nonetheless, educators appreciated that they did not need to learn many new things before they could start using the platform. The approach and platform provided domain-specific views that aligned well with their training, particularly the lesson planning features. Educators valued starting with a lesson plan, as it allowed them to focus on pedagogical goals without getting caught up in details. However, they also noted that they no longer strictly rely on formal structures for planning lessons, preferring to use their experience. This highlights the need for the platform to balance simplified lesson plans with sufficient documentation of pedagogical goals, ensuring developers have the background knowledge required to meet educators' needs. The feature to specify the type of phase for each teaching phase received mixed feedback. Educators found it a blend of two independent concepts: phase type and social interaction. The phase type should provide structure a rough structure of a lesson (e.g., motivation, exercise, discussion, conclusion). The social interaction specifies if students work on their own, in pairs, or groups. The phases which were reused from the Learning Designer (Section 2.5) were not sufficient. This could be due to limited understanding of the intended purpose and misusing the categories defined by the Learning Designer which ultimately lead to confusion with educators. Overall, the platform's usability was rated as average, with most responses close to 3.00 on a scale from 1 to 5. Interviews revealed that educators were sometimes confused by the user interface, indicating a need for user interface improvements in structure and clarity. User interface design often requires many iterations to achieve optimal results. Educators and developers were not involved in the initial development of the platform. Future development iterations could benefit from their involvement. The feedback aligns with the findings of the comparison framework, emphasizing the need for clear instructions on what to do and how to do it. The approach currently does not enforce the use of any standard techniques or notations which offers users a lot of freedom. Future iterations might benefit from more guidelines and examples to help users with applying the approach.

Educators agreed that reusability is a key requirement for effective e-learning platforms. They want to reuse lesson plans and building blocks easily. The current implementation of building blocks is limited, allowing only text-based input and requiring developer intervention before they can be used. The platform offers educators the freedom to request customized building blocks that may not already exist. This contrasts with other platforms like Elis or Moodle, which provide customizable pre-made building blocks which can be immediately used without a developer. Educators are used to creating their own learning materials and courses independently often combining freely available materials and adapting them to fit their pedagogical goals. Educators felt uncomfortable requesting materials without directly influencing their implementation, reflecting a need for more autonomy and control over their instructional materials. Future iterations of the platform need to improve this aspect as it is a core value proposition of the approach. Educators should be empowered to have as much control as possible, only involving developers if necessary. Sharing background information and requirements is essential for collaboration. Educator need to have strong incentives to share their implicit knowledge with developers. This might be achieved if educators feel like they benefit from the involvement of educators.

Communication features were generally deemed sufficient. However, it was not clear how to get insights into the collaboration. For example, the platform currently has no features which allow to create tasks and assign them to users. Educators also expressed the need to track the progress of tasks. The scope of the development approach currently does not include management activities which was also uncovered by the comparison framework. Educators requested efficient communication channels and timely notifications about updates or changes because they do not want to spend too much time on the platform. Furthermore, they noted that precise formulation of text-based requirements was challenging but also agreed that text-based communication is the only possibility. The platform could provide further guidance, examples, and communication templates in future versions.

Educators also expressed requirements for future iterations to ensure the alignment of pedagogical goals and software development processes (RQ-1).

There was a unanimous belief among educators that the platform does not provide enough value currently. They indicated that it does not significantly enhance their performance and effectiveness. Long-term planning of classes and deciding where and how e-learning fits in were identified as desired features, extending beyond the existing lesson planning capabilities. The user interface would need to adapt to handle larger lesson plans to support these needs. Educators found self-check activities and quizzes particularly useful, emphasizing the value of immediate and personalized feedback and adaptive learning paths. While the platform currently lacks dedicated features for these, planned features for course analytics could provide valuable insights for educators. In the future, course analytics could be used to enable data-driven decisions about course improvements. Educators noted that the design of e-learning courses must be modern and appealing to students, or they may not engage with them. Design guidelines for developers could help to ensure that best practices are followed. Additionally, the platform should support multiple languages and other accessibility features, which are currently not considered.

The evaluation results are not representative due to the small sample size. The evaluation also did not show if the approach can be successfully used to create an e-learning course. However, several areas of improvement could be identified. Overall, future developments should focus on improving the platform's usability and clarity and extending it to cover the complete lifecycle of the collaborative development approach. Without the ability to produce a finished e-learning course, educators struggle to justify using the platform. Support for course implementation and deployment phases is crucial for the platform's success.

8. Conclusion and Outlook

The evaluation of the proposed collaborative e-learning development platform has revealed its considerable potential, yet it also underscores the significant progress still required for its maturity. The investigation focused on identifying the essential requirements for such an approach to ensure the alignment of pedagogical goals with software development processes. A successful collaborative e-learning development approach must effectively acknowledge and cater to the diverse needs of its various stakeholders. By concealing unnecessary complexities and offering domain-specific views, the approach becomes more user-friendly. This was shown by educators quickly adopting familiar lesson planning patterns. Educators particularly appreciated the general structure of the approach and platform, finding that transitioning from lesson planning to course planning felt intuitive and seamless. This natural progression underscores the approach's ability to align with existing pedagogical workflows, thereby fostering a more effective and efficient planning process. Central to this approach is the recognition that pedagogical design should be the primary focus, with all other aspects serving to support and enhance the educational experience. This aligns with the understanding that e-learning represents pedagogy empowered by technology which can be found throughout the related work. However, it is equally important that the approach does not grant excessive freedom, as this can overwhelm users. Therefore, it must include management features to facilitate and streamline collaboration in the future. The desire for educator independence emerged as a significant theme, with reusability being an important aspect. By emphasizing reusability, the approach not only reduces development overhead but also empowers educators, enhancing their ability to independently create and manage e-learning content.

A collaborative e-learning development platform was envisioned to support the development approach. Educators identified the general idea of uniting educators and developers as a key feature of the developed platform. Intuitive and simple communication methods as well as project management features are mandatory features as they ensure that all involved stakeholders can work together harmoniously and productively. Effective communication channels are crucial for bridging the gap between pedagogical goals and technical implementation. Moreover, the automated course implementation and deployment emerged as a key functionalities. Although not implemented, these Features could significantly streamline the transition from course planning to actual instructional use. This would empower educators to work intependently and focus more on teaching and less on the technical aspects.

The current implementation leaves many things to be desired. Nonetheless, the proposed development process and platform offer a promising foundation. Continued refinement and iteration are essential to fully realize the platform's potential and address any remaining challenges. The focus of future development iterations should be on supporting the complete development cycle, encompassing course implementation and deployment. This holistic approach ensures that the platform not only facilitates the planning and creation of e-learning content but also its effective delivery and utilization. To further enhance the platform's value, incorporating course analytics is crucial, as it is a highly requested feature. By providing insights into learner engagement and performance, course analytics can help educators refine their instructional strategies and improve learning outcomes. Moreover, improving the usability of the platform is essential to increase user adoption and technology acceptance. Ensuring that the interface is intuitive and accessible will make it easier for educators and developers to navigate and utilize the platform's features. Usability enhancements could involve simplifying the user interface, reducing the learning curve, and providing comprehensive support resources such as tutorials and user guides.

To further enhance the platform's functionality and usability, several additional features should be considered. One addition could be the inclusion of pedagogical usability assessments [57]. These assessments would ensure that building blocks adhere to best practices for high pedagogical usability. The platform could also include capabilities for importing and exporting content as PDF or Word documents. This feature would prevent educators from duplicating their lesson planning efforts, allowing them to easily transfer and reuse their materials across different formats and contexts. Social features such as following, rating, reviews, and roles would increase interaction on the platform, fostering a more vibrant and engaged community. Additionally, the ability to categorize and search for building blocks using keywords, preview images, and compatible technologies would significantly improve the usability of the platform. A marketplace for building blocks could further enhance the platform's utility, offering educators a repository of ready-to-use resources. Notifications were identified as essential for managing asynchronous collaboration, ensuring that all participants are aware of updates and changes in real-time. Live updates in the user interface for chat functionalities would facilitate dynamic, interactive communication between educators and developers, enhancing collaborative efforts. The process of sending attachments and images could also be simplified by allowing users to preview content directly within the app, performing virus scans, and setting limits on uploads to ensure security and efficiency. The platform also lacks authorization features for managing permissions. This includes defining who can see and perform specific actions, explicit sharing and inviting of participants, and clear assignment of tasks and responsibilities.

Considerations about the viability of the platform's business model in the real world are necessary. It remains uncertain how educators and developers end up using the platform together. Educators have strong incentives to use the platform as they benefit from e-learning courses which can be used in their instruction. However, developers do not have an intrinsic motivation to use the platform. One idea could be to present it as an open-source platform like GitHub [15] leveraging the goodwill and collaborative spirit of the developer community. This model encourages a collaborative ecosystem where enhancements and updates are driven by collective expertise and feedback. However, even with open-source projects, most developers contribute because they personally use the software or are paid to contribute. A marketplace feature with paid building blocks could provide incentives for developers to interact with the platform. Alternatively, the platform could function as an internal tool within a company that employs both educators and developers. In this scenario, the company would manage the collaboration between these stakeholders, ensuring the creation of high-quality, engaging e-learning courses. This model provides a controlled environment where the alignment of pedagogical and technical aspects can be closely monitored and refined. It also offers the potential for dedicated resources to support the ongoing development and maintenance of the platform.

To further advance the development approach and platform, several key areas of future research should be explored. One primary avenue is investigating whether the current development approach and platform can be effectively used to create complete e-learning courses. This entails not only evaluating the platform's functionality and usability but also examining how well it supports the entire course development lifecycle from initial planning to instruction. A critical aspect of this research is determining how reusable building blocks can be defined and standardized. Identifying clear criteria and frameworks for creating and categorizing these blocks will be essential. This includes exploring the potential for integrating and reusing existing learning applications, such as H5P elements or components from the Elis platform, to enhance the flexibility and richness of the e-learning content available through the platform. Another area of focus is the automation of e-learning course implementation using reusable building blocks. Research should aim to develop methodologies and tools that facilitate the seamless assembly and deployment of course components, reducing the manual effort required from educators and developers. This automation would not only streamline the course creation process but also ensure consistency and quality across different courses. Supporting developers in this context is another possible research direction. Investigating the best practices and tools that can aid developers in integrating and customizing building blocks will be vital. This includes providing comprehensive documentation, developer SDKs, and APIs that allow for the easy extension and enhancement of the platform's capabilities. By addressing these research questions, future studies can contribute significantly to the refinement and expansion of the collaborative e-learning development approach and platform.

Data Availability

The source code, documentation, deployment scripts, interview guide, interview transcripts and survey results are available in the KIT thesis repository hosted on GitLab.

Bibliography

[1] Adeoye-Olatunde et al. "Research and scholarly methods: Semi-structured interviews". In: *JACCP: Journal of the American College of Clinical Pharmacy* 4.10 (2021), pp. 1358–1367. DOI: https://doi.org/10.1002/jac5.1441. URL: https://accpjournals.onlinelibrary.wiley.com/doi/10.1002/jac5.1441 (visited on 08/06/2024).

[2] Ljupcho Antovski and Florinda Imeri. "Review of Software Reuse Processes". In: *International Journal of Computer Science Issues (IJCSI)* 10.6 (2013).

[3] Sven Apel et al. "Software Product Lines". en. In: *Feature-Oriented Software Product Lines: Concepts and Implementation*. Ed. by Sven Apel et al. Berlin, Heidelberg: Springer, 2013, pp. 3–15. ISBN: 978-3-642-37521-7. DOI: 10.1007/978-3-642-37521-7_1. URL: https://doi.org/10.1007/978-3-642-37521-7_1 (visited on 07/04/2024).

[4] Auth0. *auth0/auth0-vue*. original-date: 2016-01-05T20:16:31Z. July 2024. URL: https://github.com/auth0/auth0-vue (visited on 07/22/2024).

[5] Auth0. *Keine Kompromisse bei der Identität*. URL: https://auth0.com/de (visited on 07/22/2024).

[6] Eddie Blass and Ann Davis. "Building on solid foundations: Establishing criteria for e-learning development". In: *Journal of Further and Higher Education* 27 (Aug. 2003), pp. 227–245. DOI: 10.1080/0309877032000098662.

[7] Robert Maribe Branch. *Instructional Design: The ADDIE Approach*. Boston, MA: Springer US, 2009. ISBN: 978-0-387-09505-9 978-0-387-09506-6. DOI: 10.1007/978-0-387-09506-6. URL: http://link.springer.com/10.1007/978-0-387-09506-6 (visited on 07/26/2024).

[8] John Brooke. "SUS: A 'Quick and Dirty' Usability Scale". In: *Usability Evaluation In Industry*. Num Pages: 6. CRC Press, 1996. ISBN: 978-0-429-15701-1.

[9] bwCloud. *bwCloud - IaaS für Forschung und Lehre in Baden-Württemberg |bw-cloud.org*. URL: https://www.bw-cloud.org/ (visited on 07/22/2024).

[10] Elizabeth Charters. "The Use of Think-aloud Methods in Qualitative Research An Introduction to Think-aloud Methods". In: *Brock Education Journal* 12.2 (July 2003). Number: 2. ISSN: 2371-7750. DOI: 10.26522/brocked.v12i2.38. URL: https://journals.library.brocku.ca/brocked/index.php/home/article/view/38 (visited on 08/06/2024).

[11] OpenAPI-Generator Contributors. *Hello from OpenAPI Generator | OpenAPI Generator*. URL: https://openapi-generator.tech/ (visited on 07/22/2024).

[12] Juan Manuel Dodero et al. "Development of e-Learning Solutions: Different Approaches, a Common Mission". In: *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje* 9.2 (May 2014). Conference Name: IEEE Revista Iberoamericana de Tecnologias del Aprendizaje, pp. 72–80. ISSN: 1932-8540. DOI: 10.1109/RITA.2014.2317532. URL: https://ieeexplore.ieee.org/abstract/document/6799263 (visited on 11/14/2023).

[13] OpenID Foundation. *How OpenID Connect Works - OpenID Foundation*. Feb. 2023. URL: https://openid.net/developers/how-connect-works/ (visited on 07/22/2024).

[14] Meiyuzi Gao, Philip Kortum, and Frederick L. Oswald. "Multi-Language Toolkit for the System Usability Scale". In: *International Journal of Human–Computer Interaction* 36.20 (Dec. 2020), pp. 1883–1901. ISSN: 1044-7318, 1532-7590. DOI: 10.1080/10447318.2020.1801173. URL: https://www.tandfonline.com/doi/full/10.1080/10447318.2020.1801173 (visited on 07/27/2024).

[15] GitHub. *Build software better, together*. URL: https://github.com (visited on 07/29/2024).

[16] Google. *Was sind progressive Web-Apps? | Articles | web.dev*. de-x-mtfrom-en. URL: https://web.dev/articles/what-are-pwas?hl=de (visited on 07/22/2024).

[17] Charles R Graham. "Chapter 1.1 Blended Learning Systems". In: *The handbook of blended learning: Global perspectives, local designs* 1 (2004), pp. 3–21.

[18] H5P Group. *H5P*. URL: https://h5p.org/ (visited on 07/26/2024).

[19] Said Hadjerrouit. "Applying a System Development Approach to Translate Educational Requirements into E-Learning". In: *Interdisciplinary Journal of E-Learning and Learning Objects* 3.1 (Jan. 2007). Publisher: Informing Science Institute, pp. 107–134. ISSN: 1552-2237. URL: https://www.learntechlib.org/p/44801/ (visited on 11/14/2023).

[20] Said Hadjerrouit. "Developing Web-Based Learning Resources in School Education: A User-Centered Approach". In: *Interdisciplinary Journal of e-Skills and Lifelong Learning* 6 (2010), pp. 115–135. ISSN: 2375-2084, 2375-2092. DOI: 10.28945/1172. URL: https://www.informingscience.org/Publications/1172 (visited on 02/08/2024).

[21] Azma Abdul Hamid. "e-Learning: Is it the "e" or the learning that matters?" In: *The Internet and Higher Education*. SOLE 2001: Exploring the Evolution of E-Learning 4.3 (Jan. 2001), pp. 311–316. ISSN: 1096-7516. DOI: 10.1016/S1096-7516(01)00072-0. URL: https://www.sciencedirect.com/science/article/pii/S1096751601000720 (visited on 07/19/2024).

[22] IBI - TU Berlin. *elis Informationen*. URL: https://info.elis-public.de/ (visited on 07/25/2024).

[23] Dirk Ifenthaler. "Technologiebasiertes Instruktionsdesign". In: *Berufs-und Wirtschaftspädagogik Online: bwp@* 5 (2017), pp. 1–12.

[24] Albert Ip, Iain Morrison, and Mike Currie. *What is a Learning Object, Technically?* Tech. rep. ERIC Number: ED466594. 2001. URL: https://eric.ed.gov/?id=ED466594 (visited on 12/06/2023).

[25] Aycan Kaya, Reha Ozturk, and Cigdem Altin Gumussoy. “Usability Measurement of Mobile Applications with System Usability Scale (SUS)”. In: *Industrial Engineering in the Big Data Era*. Ed. by Fethi Calisir, Emre Cevikcan, and Hatice Camgoz Akdag. Cham: Springer International Publishing, 2019, pp. 389–400. ISBN: 978-3-030-03317-0. DOI: 10.1007/978-3-030-03317-0_32.

[26] Badrul H. Khan. “A Framework for Web-based Learning”. In: *TechTrends: Linking Research and Practice to Improve Learning* 44.3 (2000). ISSN: 8756-3894. URL: https://www.learntechlib.org/p/90327/ (visited on 12/06/2023).

[27] Badrul H. Khan. “The People—Process—Product Continuum in E-Learning: The E-Learning P3 Model”. In: *Educational Technology* 44.5 (2004). Publisher: Educational Technology Publications, Inc., pp. 33–40. ISSN: 0013-1962. URL: https://www.jstor.org/stable/44428934 (visited on 06/30/2024).

[28] Badrul H. Khan and Vinod Joshi. “E-Learning Who, What and How?” In: *Journal of Creative Communications* 1.1 (Mar. 2006). Publisher: SAGE Publications India, pp. 61–74. ISSN: 0973-2586. DOI: 10.1177/097325860500100104. URL: https://doi.org/10.1177/097325860500100104 (visited on 06/30/2024).

[29] Ferdinand Krauss and Mohamed Ally. “A Study of the Design and Evaluation of a Learning Object and Implications for Content Development”. In: *Interdisciplinary Journal of E-Learning and Learning Objects* 1.1 (Jan. 2005). Publisher: Informing Science Institute, pp. 1–22. ISSN: 1552-2237. URL: https://www.learntechlib.org/p/44863/ (visited on 12/04/2023).

[30] UCL Knowledge Lab. *Learning Designer*. URL: https://wwwapps-live.ucl.ac.uk/learning-designer/ (visited on 05/17/2024).

[31] Christoph Ladurner et al. “Entwicklung und Implementierung eines Plug-Ins und von APIs für offene Bildungsressourcen (OER)”. In: Gesellschaft für Informatik, Bonn, 2021, pp. 453–465. ISBN: 978-3-88579-701-2. URL: https://dl.gi.de/handle/20.500.12116/34751 (visited on 05/09/2024).

[32] Diana Laurillard. “A conversational framework for individual learning applied to the ‘Learning Organisation’ and the ‘Learning Society’”. In: *Systems Research and Behavioral Science* 16.2 (1999), pp. 113–122. ISSN: 1099-1743. DOI: 10.1002/(SICI)1099-1743(199903/04)16:2<113::AID-SRES279>3.0.CO;2-C. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291099-1743%28199903/04%2916%3A2%3C113%3A%3AAID-SRES279%3E3.0.CO%3B2-C (visited on 07/20/2024).

[33] Diana Laurillard. “The pedagogical challenges to collaborative technologies”. In: *International Journal of Computer-Supported Collaborative Learning* 4.1 (Mar. 2009), pp. 5–20. ISSN: 1556-1615. DOI: 10.1007/s11412-008-9056-2. URL: https://doi.org/10.1007/s11412-008-9056-2 (visited on 05/17/2024).

[34] Yanping Liu and Hong Wang. “A comparative study on e-learning technologies and products: from the East to the West”. In: *Systems Research and Behavioral Science* 26.2 (2009). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/sres.959, pp. 191–209. ISSN: 1099-1743. DOI: 10.1002/sres.959. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/sres.959 (visited on 12/04/2023).

[35] Colla MacDonald et al. "Addressing the eLearning Contradiction: A Collaborative Approach for Developing a Conceptual Framework Learning Object". In: *Interdisciplinary Journal of E-Learning and Learning Objects* 1.1 (Jan. 2005). Publisher: Informing Science Institute, pp. 79–98. ISSN: 1552-2237. URL: https://www.learntechlib.org/p/44868/ (visited on 12/04/2023).

[36] Jonas A. Montilva, Beatriz Sandia, and Judith Barrios. "Developing Instructional Web Sites – A Software Engineering Approach". In: *Education and Information Technologies* 7.3 (Sept. 2002), pp. 201–224. ISSN: 1573-7608. DOI: 10.1023/A:1020855429304. URL: https://doi.org/10.1023/A:1020855429304 (visited on 12/04/2023).

[37] Moodle. *Kurzanleitung*. URL: https://moodle.com/de/solutions/schnellstart/ (visited on 07/12/2024).

[38] Patsy Moskal, Charles Dziuban, and Joel Hartman. "Blended learning: A dangerous idea?" In: *The Internet and Higher Education*. Blended Learning in Higher Education: Policy and Implementation Issues 18 (July 2013), pp. 15–23. ISSN: 1096-7516. DOI: 10.1016/j.iheduc.2012.12.001. URL: https://www.sciencedirect.com/science/article/pii/S109675161200084X (visited on 07/19/2024).

[39] Susan Nash. "Learning Objects, Learning Object Repositories, and Learning Theory: Preliminary Best Practices for Online Courses". In: *Interdisciplinary Journal of E-Learning and Learning Objects* 1.1 (Jan. 2005). Publisher: Informing Science Institute, pp. 217–228. ISSN: 1552-2237. URL: https://www.learntechlib.org/p/44877/ (visited on 12/04/2023).

[40] Veselina Nedeva and Snejana Dineva. "Design and Development of Efficient E-learning Courses". In: *Proceedings of the 11th International conference on virtual learning*. 2013, pp. 108–115.

[41] Mark Nichols. "E-learning in context". In: *E-Primer Series* 27 (2007), pp. 1–28.

[42] Petri Nokelainen. "An empirical assessment of pedagogical usability criteria for digital learning material with elementary school students". In: *Journal of Educational Technology & Society* 9.2 (2006), pp. 178–197. ISSN: 11763647, 14364522. URL: http://www.jstor.org/stable/jeductechsoci.9.2.178 (visited on 02/08/2024).

[43] Thomas Olbrecht. *Akzeptanz von E-Learning Eine Auseinandersetzung mit dem Technologieakzeptanzmodell zur Analyse individueller und sozialer Einflussfaktoren*. deutsch. Oct. 2010. URL: https://www.db-thueringen.de/servlets/MCRFileNodeServlet/dbt_derivate_00021996/Olbrecht/Dissertation.pdf (visited on 05/22/2024).

[44] OpenAPI. *Home 2024 - OpenAPI Initiative*. URL: https://www.openapis.org/ (visited on 07/31/2024).

[45] Seymour Papert. "A Critique of Technocentrism in Thinking About the School of the Future". In: *Children in the Information Age*. Elsevier, 1988, pp. 3–18. ISBN: 978-0-08-036464-3. DOI: 10.1016/B978-0-08-036464-3.50006-5. URL: https://linkinghub.elsevier.com/retrieve/pii/B9780080364643500065 (visited on 08/06/2024).

[46] Patrick E. Parrish. "The trouble with learning objects". In: *Educational Technology Research and Development* 52.1 (Mar. 2004), pp. 49–67. ISSN: 1556-6501. DOI: 10.1007/BF02504772. (Visited on 12/06/2023).

[47] Quasar. *Quasar Framework - Build high-performance VueJS user interfaces in record time.* URL: https://quasar.dev/ (visited on 07/22/2024).

[48] William H. Rice. *Moodle: e-learning course development ; a complete guide to successful learning using Moodle.* 1st ed. From technologies to solutions. Birmingham: Packt-Publishing, 2006. ISBN: 978-1-904811-29-9.

[49] Frank Schwarz. "e-Learning in den Ingenieurwissenschaften – Entwicklung, Anwendung und Evaluation einer internetbasierten Lernumgebung". PhD thesis. Duisburg, Essen, Univ., 2009.

[50] Caddy Web Server. *Reverse proxy quick-start - Caddy Documentation.* URL: https://caddyserver.com/ (visited on 07/22/2024).

[51] Samuel Ssemugabi. "Usability evaluation of a web-based e-learning application: a study of two evaluation methods". PhD thesis. University of South Africa, 2006.

[52] Ruoyu Su and Xiaozhou Li. *Modular Monolith: Is This the Trend in Software Architecture?* arXiv:2401.11867 [cs]. Jan. 2024. URL: http://arxiv.org/abs/2401.11867 (visited on 07/22/2024).

[53] VMware Tanzu. *Spring Boot.* URL: https://spring.io/projects/spring-boot (visited on 07/22/2024).

[54] Maryam Tayebinik and Marlia Puteh. "Blended Learning or E-learning?" In: *International Magazine on Advances in Computer Science and Telecommunications* (2013), pp. 103–110. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2282881.

[55] Kelvin Thompson and Francisca Yonekura. "Practical Guidelines for Learning Object Granularity from One Higher Education Setting". In: *Interdisciplinary Journal of E-Learning and Learning Objects* 1.1 (Jan. 2005). Publisher: Informing Science Institute, pp. 163–179. ISSN: 1552-2237. URL: https://www.learntechlib.org/p/44874/ (visited on 12/04/2023).

[56] Venkatesh and Davis. *A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies | Management Science.* 2000. URL: https://pubsonline.informs.org/doi/abs/10.1287/mnsc.46.2.186.11926?casa_token=A3tfMhWZCCkAAAAA:KLq6m_1FdRM3du4gw655hRD6h42VkP0p38JRwm29SbiYjwdzP6YU7N3rwV--jN7uKZOI4EtoblQ (visited on 07/25/2024).

[57] Robert Wadholm et al. "Assessing the Pedagogical Usability of a Digital Resource Sharing Library". In: *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education.* 2013, pp. 208–214. URL: https://www.academia.edu/7672331/Assessing_the_Pedagogical_Usability_of_a_Digital_Resource_Sharing_Library (visited on 02/08/2024).

[58] Deryn M. Watson. "Pedagogy before Technology: Re-thinking the Relationship between ICT and Teaching". In: *Education and Information Technologies* 6.4 (Dec. 2001), pp. 251–266. ISSN: 1573-7608. DOI: 10.1023/A:1012976702296. URL: https://doi.org/10.1023/A:1012976702296 (visited on 12/18/2023).

A. Appendix

A.1. Lesson Plan Structure

Strukturierungs*vorschlag* für die Ausarbeitung eines ausführlichen Unterrichtsentwurf (8-10 Seiten)

1. Deckblatt
Name, Art der Veranstaltung, Thema, Datum, Ort, Uhrzeit, …

2. Rahmenbedingungen
- Merkmale der Gruppe: Größe, Zusammensetzung (homogen/heterogen; berufsspezifisch/unspezifisch; Altersunterschiede; Geschlecht; …), Gruppenbildung (freiwillig/Pflicht/…)
- Merkmale der Lernenden (Vorwissen; Lernstrategien, evtl. Arbeitsverhalten u. Motivation
- Räumliche u. materielle Situation (Räume, Medienausstattung, …)
- (evtl.: besondere Bedingungen der Institution: Prüfungsanforderungen; Vorgaben von Vorgesetzten bzw. curriculare Vorgaben ← dies kann auch unter Punkt 4 erfolgen)

3. Sachanalyse
- muss nicht aufgeführt werden

4. Didaktische Überlegungen (mikrodidaktische Ebene)
- Welche Inhalte vermittle ich?
  → didaktische Reduktion: Welches sind die wichtigen Inhalte? (Schwerpunktsetzung); Die Auswahl und Begründung ist in Bezug auf die Lernenden und eventuellen institutionellen Vorgaben vorzunehmen
  Mögliche Leitfragen: Wie ist das Vorwissen (Interesse/Erwartungen) der Teilnehmer/innen? Was können die Lernenden innerhalb des zeitlichen Rahmens verarbeiten? Hat der Lerninhalt eine exemplarische Bedeutung? Bietet er Transfermöglichkeiten? Gibt es curriculare Vorgaben?
- Aus diesen Überlegungen sollte die Formulierung der Lernziele hervorgehen.
- Handelt es sich um keine abgeschlossene Einheit, sollte kurz angeführt werden, was bisher behandelt wurde bzw. was nachfolgt.

5. Lernziele
- Neben rein inhaltlichen (kognitiven) Zielen sollten, sofern in der Veranstaltung angestrebt, auch gefühlsbezogene (affektive) und/oder verhaltensbezogene (psychomotorische) Ziele aufgeführt werden. Die Formulierung der Ziele kann sich auch an den Bereichen Sach-, Methoden und Sozialkompetenz orientieren.
- Vorschlag: Zunächst ein allgemeines Gesamtziel für die Veranstaltung formulieren und dann einige Teilziele anführen.

6. Methodische Überlegungen
- Wie vermittle ich die Inhalte?
  Auswahl und Begründung des methodischen Vorgehens in Bezug auf die Ziele/Inhalte und die Teilnehmer:
  Strukturierung der Veranstaltung (Sandwich-Prinzip?); Auswahl der Sozialformen; Einsatz spezifischer Methoden und Medien; Gestaltung der Einstiegs/Abschlussphase; Diskussion alternativen Vorgehens; Planung didaktischer Weichen bzw. Offenhalten des Vorgehens; Berücksichtigung des zeitlichen Rahmens

7. Verlaufsplanung
- Unterrichtsskizze/Raster (viele Formen möglich): z.B.:
  Zeit / U-phase/Lehrform / Sozialform / Medien / Bemerkungen …

8. Literaturangaben

9. Anhang (Teilnehmerunterlagen, …)

Figure A.1.: Lesson Plan Structure by Pädagogische Hochschule Karlsruhe

A.2. Deployment Scripts

```
<insert application domain> {
  redir / /public/index.html 308

  reverse_proxy elst:8500
}
```

Figure A.2.: Caddy Configuration

```
FROM node:20 as frontendBuildStep

COPY frontend /frontend
WORKDIR /frontend

RUN npm ci
RUN npm run build

FROM maven:3.9.6-amazoncorretto-21 as backendBuildStep
MAINTAINER baeldung.com

COPY backend /backend
WORKDIR /backend

COPY --from=frontendBuildStep /frontend/dist/pwa src/main/resources/static

RUN mvn package && mv target/elst-*.jar target/elst.jar


FROM amazoncorretto:22-alpine-jdk

COPY --from=backendBuildStep /backend/target/elst.jar elst.jar

ENTRYPOINT ["java","-jar","/elst.jar"]
```

Figure A.3.: Backend Dockerfile

```
version: "3.3"
services:
  caddy:
    image: caddy:2.7-alpine
    restart: unless-stopped
    ports:
      - "80:80"
      - "443:443"
      - "443:443/udp"
    volumes:
      - ./Caddyfile:/etc/caddy/Caddyfile
  elst:
    build: .
    environment:
      DB_USERNAME: "root"
      DATABASE_URL: db:3306
    ports:
      - "8500:8500"
  db:
    image: mysql:8.0
    restart: always
    environment:
      MYSQL_DATABASE: "elst"
      MYSQL_USER: "elst"
      MYSQL_PASSWORD: "<insert password>"
      MYSQL_ROOT_PASSWORD: "<insert password>"
    expose:
      - "3306"
    volumes:
      - my-db:/var/lib/mysql
volumes:
  my-db:
```

Figure A.4.: Docker Compose Script

A.3. OpenAPI Specification

Figure A.5.: Backend API Specification using OpenAPI

Figure A.6.: Backend API Specification using OpenAPI

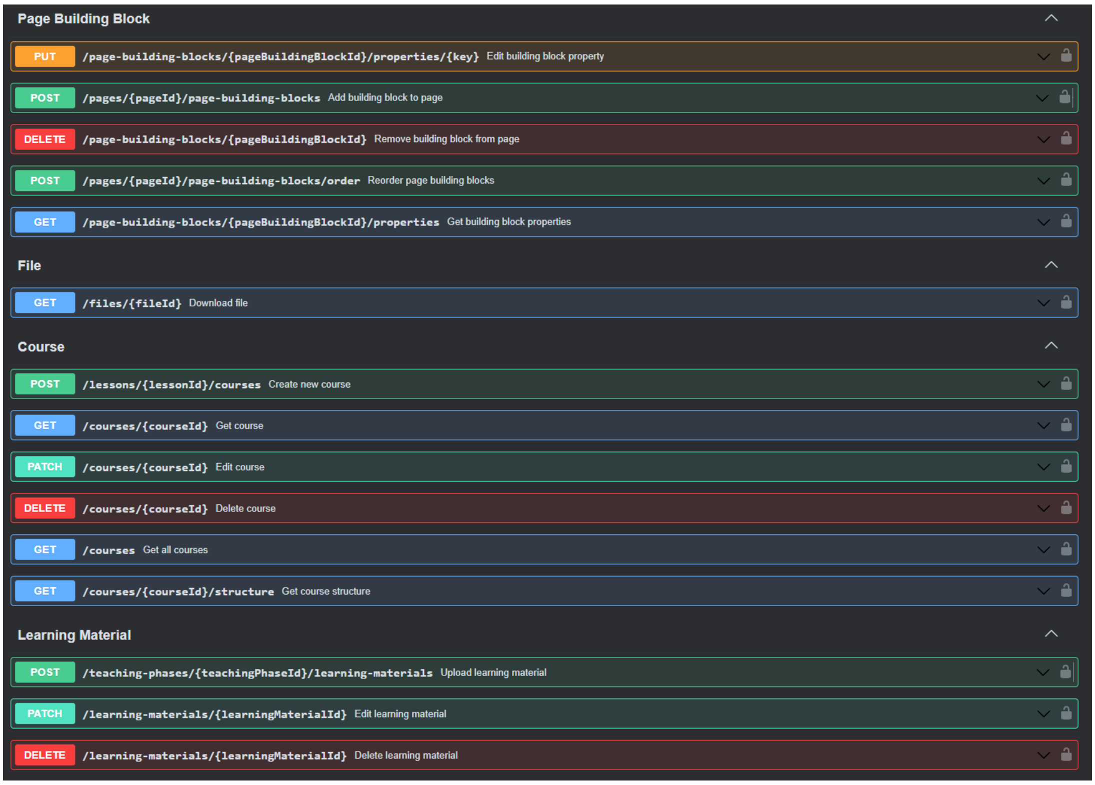

Figure A.7.: Backend API Specification using OpenAPI

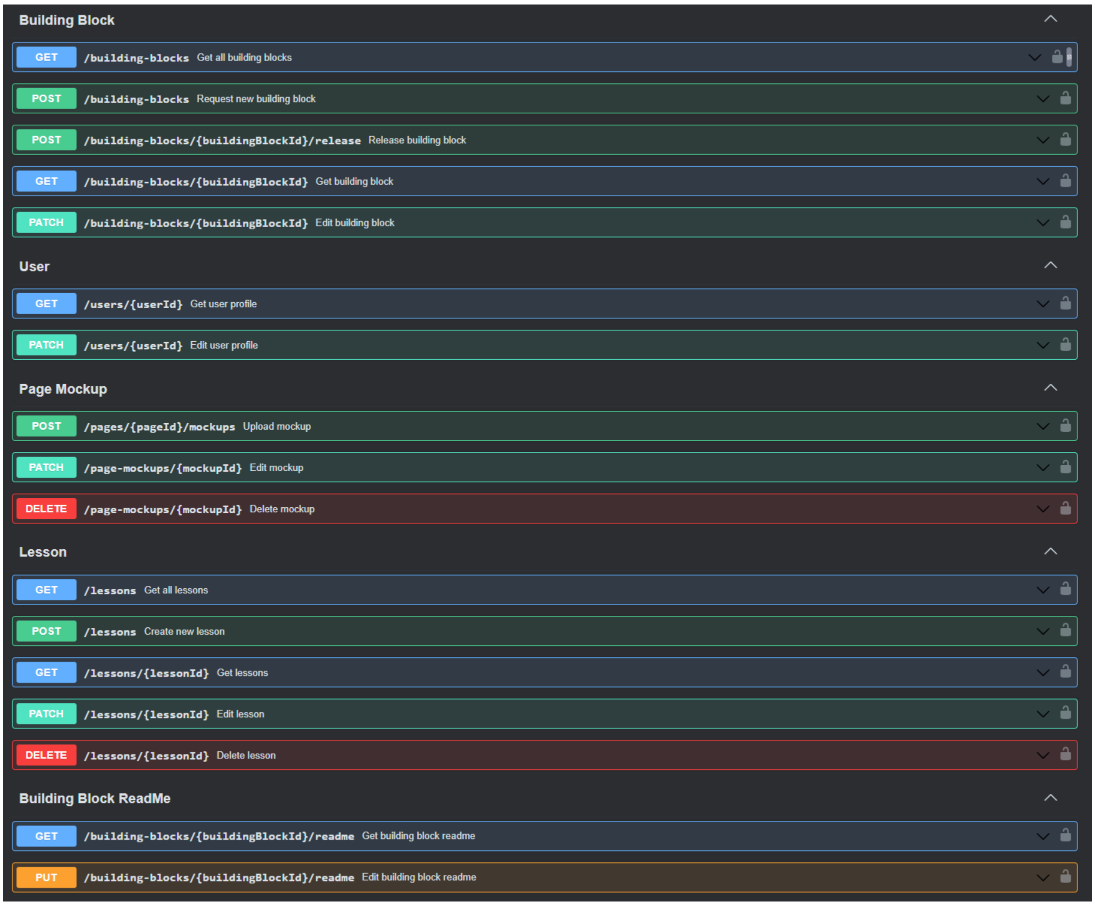

Figure A.8.: Backend API Specification using OpenAPI