Study of design thinking and software engineering integration in education and training

Muhammad Ihsan Zul[1], Suhaila Mohd. Yasin[2], Dadang Syarif Sihabudin Sahid[1]
[1]Department of Information Technology, Politeknik Caltex Riau, Pekanbaru, Indonesia
[2]Department of Software Engineering, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Batu Pahat, Malaysia

Article Info

Article history:

Received Oct 7, 2024
Revised Apr 10, 2025
Accepted Jul 2, 2025

Keywords:

Design thinking
Education
Integration
Software engineering
Training

ABSTRACT

Integrating design thinking (DT) with software engineering (SE) is widely applied in industry, serving as a reference for SE in education and training. The industry has various integration models, but researchers and educators mainly adapt them for education. A clear understanding of DT-SE integration models is essential to figuring out their implementation. This study examines existing DT-SE integration models, challenges, and integration methods using Kitchenham's framework in education and training. The paper was collected from ScienceDirect, IEEEXplore, Scopus, ACM, SpringerLink, and Google Scholar, yielding 593 initial publications, with 43 selected for in-depth analysis. Findings indicate that the d.school model is the most widely adopted DT model. Key challenges include team dynamics, process management, complexity, and cultural factors. DT is integrated into requirements engineering (RE) due to its user-centered nature, though only two studies explicitly describe DT-SE integration models, both applied early in SE processes. These findings suggest educational practices align with industry trends in model adoption and integration focus. Educators and practitioners can use these insights to design or adapt integration models suitable for education and training by shaping curricula that emphasize user-centered design, collaboration, and the extension of DT practices beyond RE-strengthening its impact for education and training.

Corresponding Author:

Muhammad Ihsan Zul
Department of Information Technology, Politeknik Caltex Riau
Pekanbaru, Indonesia
Email: ihsan@pcr.ac.id

1. INTRODUCTION

Design thinking (DT) is an innovative approach that has gained significant traction across various sectors [1], including industry and education. It is frequently employed as a strategy for fostering innovation, serving as a solution-oriented methodology to address challenges in diverse fields, including those organizations face [2]. In the educational realm, DT has been successfully implemented across a range of disciplines, with software engineering (SE) standing out as a particularly notable example [3], [4]. Within SE, DT designs and develops impactful software solutions. The concept of "impactful" in this context refers to the creation of products that are not only innovative but also closely aligned with the specific needs and expectations of end-users. The adoption of DT in SE has resulted in the development of various software products, such as student information systems [5] and digital banking [6], among others, all aimed at enhancing human activities within the increasingly digital landscape.

This adoption is often facilitated through various learning schemes, such as workshops or training sessions, designed to equip teams with the necessary skills to implement DT effectively [7]. By observing DT's successful application in industry, educational institutions have increasingly sought to incorporate it into their curricula [8] through teaching activities [9]. This trend aims to prepare students with the requisite knowledge and skills for DT application, ensuring they are well-equipped to contribute to future innovation efforts [8], [10]. Both industry and education sectors share a common goal in their initial adoption of DT: the development of skilled design thinkers who can produce human-centered innovations. Despite the growing interest in DT, investigation research on its integration with SE in education and training remains underexplored. There is a need for a more structured study to incorporate DT into SE education and training. While DT is acknowledged for fostering innovation and problem-solving, its application in SE education and training should be mapped to provide a clearer understanding of integration from existing studies.

Systematic literature reviews (SLR) on SE and education have been widely conducted, with many studies focusing on applying methodologies for software development, particularly within industrial contexts. On the other hand, research focusing on implementing DT in SE education often emphasizes user experience (EX) design. Such studies explore how DT can enhance students' skills within a project-based learning (PBL) framework. For instance, the work by Ferreira and Canedo [11] examines the application of DT using Google's Design Sprint model in project based learning settings for UX design. The research indicated that the approach enhanced student engagement and UX quality. However, it is important to note that this study focused solely on student activities utilizing Google's Design Sprint models and did not explore other DT models. Next, a study that delves into the integration of DT with agile software development (ASD) has been conducted. The study examines 29 papers to identify the DT model, challenges, and how to integrate DT with ASD [12]. This study highlighted various DT models in ASD, including ISO, IDEO, d.school, Google Design Sprint, and Double Diamond, alongside ASD models like XP, Lean, Kanban, and Scrum. Moreover, they emphasize the challenges during the integration, such as geographical differences, tight schedules, limited budgets, communication barriers, and resource allocation. This study focused on industry context and did not examine the application in an educational and training context.

Similar to previous studies, Parizi *et al.* [13] investigate 127 papers that use DT within software development more broadly, addressing integration strategies such as Upfront DT, Infused DT, and Continuous DT, as proposed by [14]. This study provides valuable insights into DT techniques and selection criteria for software development, but it did not explicitly discuss the application of DT in educational or training contexts. In an education setting, the action research study compares ASD and DT approaches in a SE education setting [15]. In this study, two student teams employed Agile and DT methodologies over 16 sessions within a regular course framework. The results suggest that Agile approaches are quicker, while DT offers deeper engagement with development uncertainties and fosters creativity and innovation. The results of this study indicate the critical role of DT in learning. However, the findings need to explore further the different DT models used, as varying stages can significantly impact project completion speed.

Given the current circumstances, exploring how DT can be effectively integrated into SE in education and training is essential. Identifying the most widely used DT models combined with ASD is a key step. Understanding challenges in DT implementation-both external and internal-will help improve its application in learning environments. Since DT can be applied at various stages of SE, optimizing its use in software development is crucial. Therefore, the primary aim of this research is to comprehensively study the application of DT and SE within education and training, covering models, challenges, and the proposed integration models. Ultimately, this study contributes as a reference for both educators and practitioners implementing or developing DT integrated with SE in education and training settings. Aside from this, this study supports researchers in expanding future research efforts in DT integrated with SE towards other potential directions, such as product evaluation and security analysis.

This research is systematically structured to ensure clarity and coherence. The introduction presents the research problem, includes the study's objective, and outlines its contribution by reviewing relevant previous studies. The methodology section explains the approach and stages of this study. The results and discussion section provides detailed findings and analysis at each stage, thoroughly interpreting the results and comparing them with similar studies. Finally, the conclusion section highlights the research contributions and suggests directions for future work.

2. METHOD

The research utilized Kitchenham's framework for SLR, a method acknowledged for its systematic approach to synthesizing academic findings. This methodology encompasses three pivotal phases: planning the review, conducting the review, and reporting the review [16], [17]. Initially, the planning phase involves formulating research questions, elaborating a detailed methodological strategy, determining inclusion and exclusion criteria for studies, and selecting digital libraries for the literature search. This stage is crucial for establishing a clear and methodologically sound foundation for the review. After this, the conducting phase comprehensively searches related literature within the selected digital libraries, employing a systematic selection process based on predefined quality criteria and extracting relevant data from the identified papers. This phase is essential for selecting literature into a focused subset of pertinent research. The final stage of the process, the reporting phase, synthesizes the findings, addressing the research questions while providing written and visual explanations of the outcomes. It also includes an extensive analysis of the review results, a discussion of the limitations encountered, and proposes directions for future research. This meticulous approach ensures the thoroughness and reliability of the study’s outcomes.

The methodology section will thoroughly explain the comprehensive stages of planning. The results section will discuss the conducting review and reporting stages in detail. The research stages based on are illustrated in Figure 1.

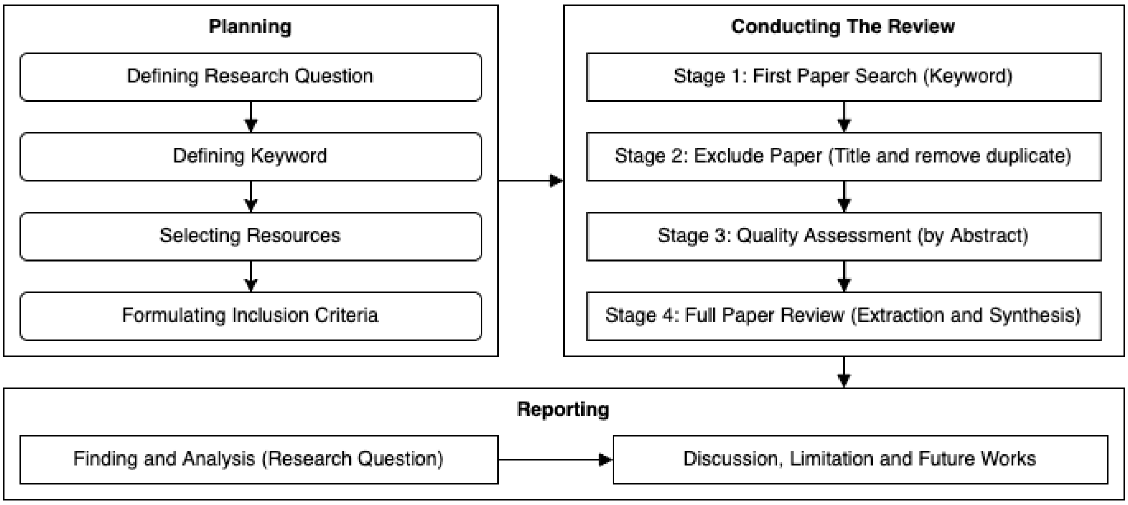

Figure 1. Research framework adopted from Kitchenham’s framework

2.1. Research questions

This research explores DT and SE integration in education and training by examining existing DT models and integration models and identifying challenges. It seeks to use these findings to propose new models to improve SE through DT, specifically within education and training. This exploratory approach will not only assess current integration strategies but also contribute to the development of innovative methods to enhance the quality of SE with DT support. Based on the research aims, the research questions proposed in this study are:

- RQ1: what is the most commonly used DT model applied in SE education and training?
- RQ2: what challenges are encountered in integrating DT with SE in education and training?
- RQ3: what are the stages where DT is utilized in SE education and training?
- RQ4: which research shows the DT and SE integration model in their works?

2.2. Keywords and resources

A set of specific keywords was carefully selected to conduct the literature search and ensure a focused and comprehensive retrieval of relevant studies. The chosen keywords included “design thinking,” “software development,” “software engineering,” “education,” “software engineering course,” and “training,” covering a broad range of topics related to integrating DT and software engineering in education and training.

The research utilized a variety of leading academic databases and digital libraries, including Scopus, Google Scholar, SpringerLink, Elsevier Science Direct, IEEE Xplore, and the Association for Computing Machinery (ACM). These sources were chosen for their comprehensive collections of peer-reviewed journals, conference papers, and scholarly publications, ensuring the literature review was based on rigorous and pertinent research. This diverse range of databases also aimed to encompass a broad spectrum of insights and discoveries in the field.

2.3. Inclusion criteria

The selection criteria for papers included in this study were designed to ensure that only the most relevant and high-quality research was considered. First, the papers needed to focus on research within SE, software development, ASD, and DT. This criterion ensured that the studies were related to integrating these methodologies within educational and training contexts. Second, the selected studies had to specifically relate to education and training, ensuring the research applied to enhancing educational frameworks and training programs. Third, only research papers written in English were considered, specifically those published as journal articles, conference papers, or book chapters, to maintain a consistent language and format for analysis. Lastly, the papers must be published within the last five years, between 2020 and 2024, to ensure that the study was based on the most recent advancements and trends. This time frame was selected to reflect the current research and practices in DT and SE as applied to education and training.

A systematic process was followed during the second stage, "conducting the review" to ensure that all papers met the specified criteria [16]. This process involved several steps to guarantee that only the most relevant and high-quality research was included in this study: 1) evaluating the relevance of each paper based on its title and eliminating any duplicates to maintain unique papers. At this stage, paper titles were selected in detail to see if they contained elements of DT application with SE 2) performing a quality assessment by analyzing the abstracts to determine their appropriateness for the study. The part of the reviewed abstract includes the research objective, method, and results, and 3) thoroughly reading the full text of the selected papers to obtain a deeper and more detailed understanding of each study. At this stage, a thorough study was conducted on whether the papers discussed the DT model, the challenges faced in integration, in which SE stage DT was applied in their studies, and whether or not they depicted their integration model visually.

2.4. Reporting

The reporting section represents the final stage of this research. This section will extract and organize information about the research questions, including tables and charts. The study is conducted comprehensively for all selected papers. The primary reference for the DT model is the study by Pereira and Russo [12]. Relevant references include the SE stages and integration strategies for the implementation of DT in the SE stage [14]. Detailed reporting results are presented as answers to all RQs.

3. RESULTS AND DISCUSSION

This section presents the results obtained during the conducting the review and reporting stages of the Kitchenham framework.

3.1. Conducting the review result

In this stage, a search was performed using targeted keywords across each chosen digital library. Alongside using keywords, filtering options were applied within each digital library according to the features available on the respective platforms. These filtering options included selecting the type of papers, such as review articles, conference papers, or journal articles, with all publications specifically related to computer science. The results from this initial search are shown in Table 1.

Table 1. First search result

Source	Constraints	Keyword	Result (papers)
Science Direct	– Review and research articles, book chapters – Computer science subject areas	"Design thinking" AND "software engineering" AND ("education" OR "training")	44
Google Scholar	– Review articles	"Design thinking" AND "software engineering" AND ("education" AND "training")	262
IEEE Xplore	– Conferences and journals	("All metadata": Design thinking) AND ("all metadata": Software Engineering) AND (("All Metadata": education) AND ("all metadata": training))	109
Scopus	– Computer science subject areas – Conferences and journals	"Design thinking" AND "software engineering" AND ("education" OR "training")	18
SpringerLink	– Article and research article – Computer science subject areas	"Design thinking" AND "software engineering" AND ("education" OR "training")	23
ACM	– Journal and proceeding	"Design thinking" AND "software engineering" AND ("education" OR "training")	137

The initial search yielded many selected papers, with Google Scholar contributing the most results (262), followed by ACM (137) and IEEE Xplore (109). In Stage 2, the relevance of the selected papers from each digital library was meticulously evaluated. Relevance was determined by examining whether the titles incorporated the specified keywords or their synonyms. Papers not meeting these criteria were excluded, eliminating 316 papers irrelevant to the research. Google Scholar produced the most irrelevant results of these eliminated papers, followed by ACM and IEEE Xplore. Additionally, duplicate publications across different digital libraries were identified and removed. Several publications were found to be duplicated in both Google Scholar and Scopus, as well as in other digital libraries. The outcomes of the paper filtering process in Stage 2 are shown in Table 2.

The next step involved filtering papers in Stage 3. In this stage, papers were selected by assessing their quality based on their abstracts. 153 papers were analyzed and evaluated for quality by examining the clarity of their research objectives. Overall, the selected papers were high quality and suitable for inclusion in this study. However, regarding relevance to the study’s objectives and methodology, some papers were deemed not applicable. Specifically, it was identified that certain papers did not explicitly address DT but instead focused on approaches similar to DT, such as user-centered design, human-centered design, and lean startup. Additionally, some papers should have discussed SE activities within the context of education and training, falling outside this research’s scope. The results of the Stage 3 are presented in Table 3 (Stage 3).

After selecting 56 papers, the next step involved extracting and synthesizing information by thoroughly reading the full content of each selected paper. This phase represents the final stage of the conduct the review process, where further elimination of papers that need to be more relevant based on content synthesis is still possible. Initially, 56 papers were selected, but 43 were chosen as key references for this study, excluding 13. The excluded papers generally did not focus on DT as the primary topic of their research, only briefly mentioning DT within their discussions. The results of the paper selection process in this stage are presented in Table 3 (Stage 4).

In conclusion, the conducting the review phase began with an initial search that identified 593 papers based on predefined keywords. Following the first stage, which involved reviewing titles and removing duplicates, 153 papers were selected for further analysis. The second stage entailed a quality assessment by evaluating the abstracts, resulting in the selection of 56 papers. The final stage involved a comprehensive review of the full content of these papers, during which information was extracted and synthesized to assess each paper’s contribution to the research. Ultimately, 43 papers were chosen as key references. This process ensured that only the most relevant and high-quality sources were included in this study. The overall result for each stage in conducting the review phase is shown in Figure 2.

Table 2. Stage 2 result

Source	Duplicate	Exclude	Include	Total
Science Direct		17	27	44
Google Scholar	13	216	33	262
IEEE Explore		64	45	109
Scopus	4	10	8	18
SpringerLink		9	14	23
ACM		111	26	137
Total	17	427	153	593

Table 3. Stage 3 quality assessment and stage 4 extraction and synthesis

Source	Stage 3 quality assessment			Stage 4 extraction and synthesis		
	Exclude	Include	Total	Exclude	Include	Total
Science Direct	19	8	27	2	6	8
Google Scholar	29	4	33	3	1	4
IEEE Explore	21	24	45	5	19	24
Scopus	-	8	8	1	7	8
SpringerLink	10	4	14	2	2	4
ACM	18	8	26		8	8
Total	97	56	153	13	43	56

3.2. Papers descriptive analysis

As discussed in section 3.1, the number of papers selected for this study is 43. Among these are 14 journal articles, 28 conference papers, and 1 book chapter. Additionally, the types of study conducted in all selected papers are illustrated in Figure 3.

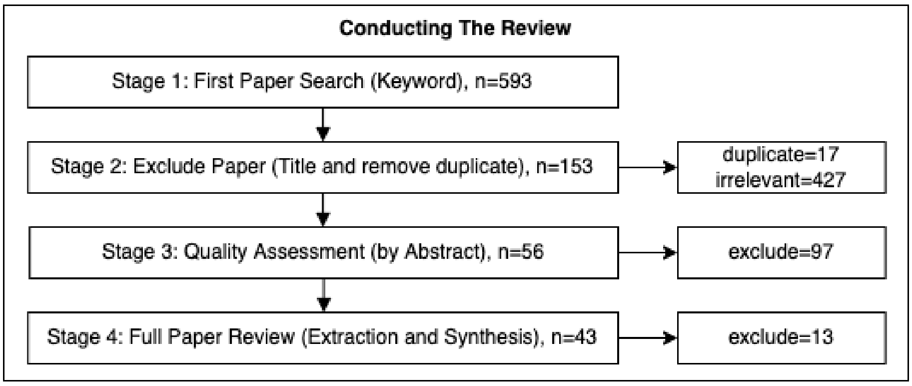

Figure 2. Conducting the review result

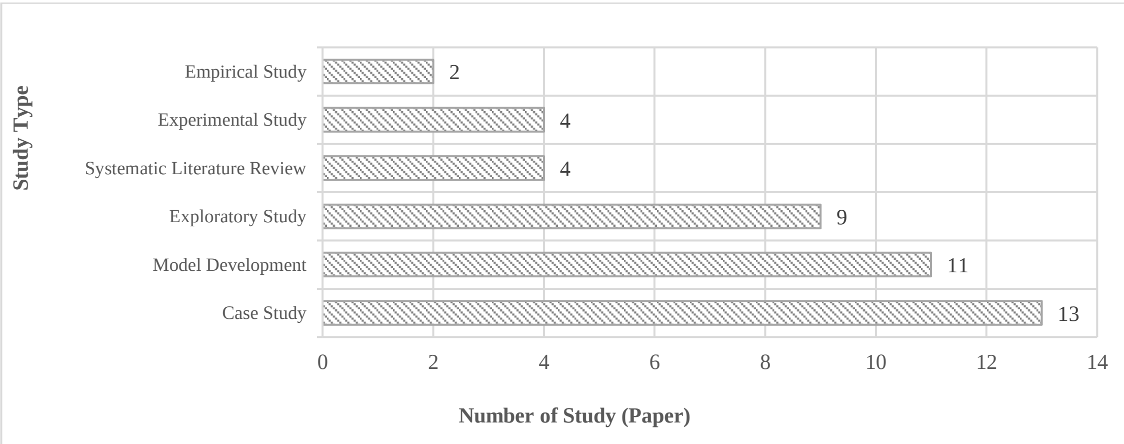

Figure 3. Study type

Based on Figure 3, it can be observed that case study and model development research types are the most frequently selected in this study. This aligns with the study's objective of integrating DT with SE. Furthermore, the distribution of publication years for the selected papers is shown in Figure 4. The figure indicates that only one paper from the year 2024 was selected, as opposed to the previous years, which showed an upward trend for both journal and proceeding articles. Therefore, research in this domain has great potential to be explored.

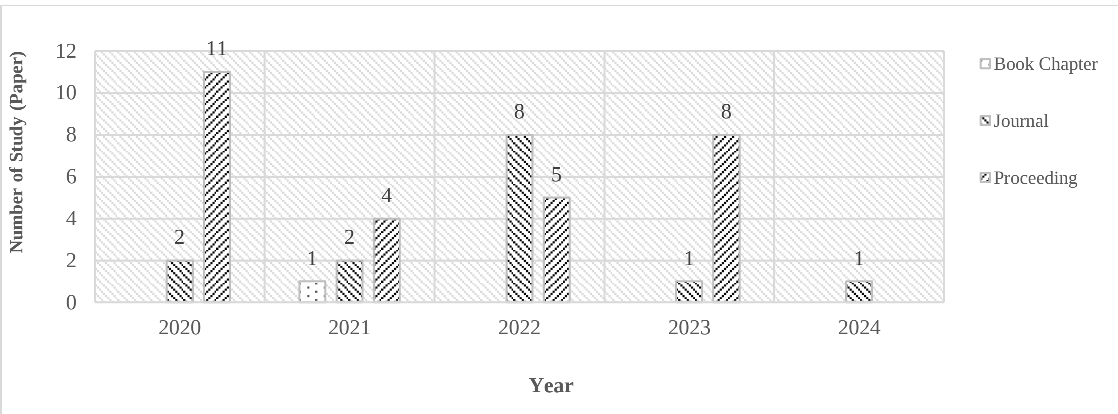

Figure 4. Paper by year

3.3. Research question

3.3.1. RQ1 what is the most commonly used DT model applied in SE education and training?

Based on the findings by Russo and Pereira [12], at least five DT models are widely utilized today. Large companies such as IBM and SAP have also introduced their DT models. Therefore, the most frequently used DT models in the 43 research papers reviewed can be seen in Table 4.

Table 4. Design thinking model

Design thinking model	Phases and activities	Reference
IDEO by international design and consulting firm	Inspiration, ideation, and implementation	[12], [18]
Institute of design at Stanford (d.school)	Empathize, define, ideate, prototype, test	[12], [18]
International organization for standardization (ISO): human-centered design for interactive systems	Understand, specify, produce, evaluate	[12], [18]
Google design sprint	Understand, sketch, decide, prototype, validate	[12], [18]
Austin center for design, educational program	Ethnography, synthesis, prototyping	[12]
Design council UK, double diamond	Define strategy (understand, define) and execute solution (explore, create)	[12]
IBM design thinking	Observe, reflect, make	[18]
SAP	Plan, research, design, adapt, measure	[19]

Of the 43 papers, 11 did not explicitly mention the DT model they used in their research. Although these papers did not specify the DT model, it is possible to identify it based on the DT phases mentioned in the studies. However, these papers did not explain the detailed DT phases of their research. As a result, these 11 papers were categorized as not explicitly identifying the DT model used. Furthermore, the breakdown of DT models used in the remaining 32 papers is as follows: 1) d.school in 25 papers, 2) combination models in 3 papers, 3) Google Design Sprint in 2 papers, 4) Double Diamond in 1 paper, and 5) IDEO in 1 paper. Therefore, the most widely used model in SE for education and training is the d.school model. This result is confirmed by a prior study [18] based on their investigation through the survey approach in a particular region; the most used model in SE is d.school.

3.3.2. RQ2 what challenges are encountered in integrating DT with SE in education and training?

The challenges were identified by detecting the issues raised in each paper reviewed. These challenges were then collected and categorized based on their semantic similarities, leading to the grouping of challenges into several main categories: team, process and management, complexity, and culture. These main challenges were structured according to the sub-challenges discussed across the papers. Notably, some papers explicitly mentioned more than one challenge within their research. The details of the main challenges and their subcategories are presented in Table 5.

Table 5. Main challenge categories

No	Main challenges	Challenges	Paper
1	Complexity	Managing complexity, technical complexity, managing complex project dynamics, diverse needs and objectives	[6], [20]-[27]
2	Culture	Cultural shifting, cultural issues, cultural barriers, mindset change, different values, different stakeholders, organization resistance, fostering culture, gap business value and management	[7], [23], [28]-[31]
3	Process/ Management	Resource allocation, scheduling, balancing activities, misalignment, balancing creativity and iterative process, technical constraints, gap business value and management Implementation issues, online and offline activities, tools, documents, lack of user interaction	[29], [30], [32]-[43]
4	Team	User research issues, team understanding, unclear clients need, poorly written documents, miscommunication, inconsistent activities, collaboration, failure to capture user needs and expectations, team dynamics, difficulty in aligning team ideas, technical barriers between developer	[5], [11], [21], [22], [27], [28], [30], [32], [33], [43]-[58]

Figure 5 reflects the order of challenge categories by the number. The order is papers, team, process/management, culture, and complexity. It can be concluded that the primary challenge in integrating DT with SE in education and training lies within the team. The sub-challenges related to the team involve the ability of members to apply DT when combined with SE practices effectively. These challenges encompass crucial aspects such as communication, collaboration, understanding, team dynamics, and documentation. The team’s ability to navigate these areas is vital to the success of DT-SE integration, highlighting the importance of strong team dynamics in the educational context.

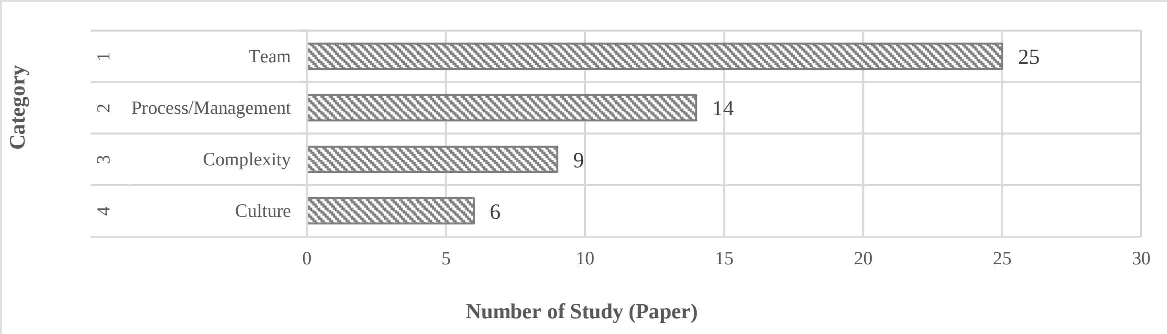

Figure 5. Challenges based on paper

Compared to research on requirements engineering (RE) in SE in education [17], it is evident that client orientation, interviewing, collaboration, and communication skills are essential for teams, particularly for RE engineers. These skills align closely with the challenges identified in this study, especially those related to team functionality. The necessity of these skill sets poses a significant challenge when implementing the integration of SE and DT, underscoring the critical importance of team competence in this context. Therefore, studies focused on strengthening team capabilities in applying DT-SE in education and training not only address current challenges but also present a promising area for future research development in this field.

Several studies have proposed various solutions to the challenges outlined in Table 5. In the context of team dynamics, particularly in collaboration and communication, the study [22] developed multiple communication tools and reminder cards to enhance the quality of team collaboration. This innovation ensures all team members remain connected and aligned in their tasks. Similarly, a study [44] integrated SE activities with DT in a problem-based learning (PBL) model. This approach allows for directly applying DT principles within SE, ensuring that software development is more structured and focused on creative solutions. Additionally, DT plays a role in the evaluation process to ensure that students acquire technical skills and continue to develop in line with industry needs [32]. In terms of process and management, various approaches have been developed. Tools such as a DT toolbox [30] and a persona-based assessment framework [32] are crucial to helping teams manage SE processes effectively. Furthermore, the development of the discovery effort worthiness (DEW) Index [35] and the creation of a digital innovation environment contribute to balancing the application of SE and DT [36]. Combining DT with other approaches helps create a more adaptive environment and effective project management.

The complexity of software development is addressed not only with advanced management tools but also by utilizing AI to support automation and faster decision-making [21]. Developing better collaboration tools and DT models integrated with co-creation methods enhances user involvement in the development process [6], [27]. A “design for real people” approach was created to ensure the team remains focused on end-user needs [25]. Addressing this complexity is further facilitated by aligning organizational values with individual team members’ values, fostering greater collaboration within the development team [29]. On the other hand, cultural challenges have become the last challenges in this study. Research indicates that one-day workshops focused on DT application for professionals are an effective strategy for addressing cultural issues in the workplace [7]. A DT integrated with a SE framework is also used as a guideline for handling cultural challenges [23], [29]. The fundamental solution to these issues is ensuring software developers adapt to the DT model more profoundly and comprehensively, fostering a culture of innovation and collaboration within the organization [31].

3.3.3. RQ3 what are the stages where DT is utilized in SE education and training?

Identifying DT application within SE stages in education and training was done by thoroughly examining how DT was implemented and the objectives achieved in the reviewed studies. For studies that depicted the integration of DT with SE, the identification process involved analyzing the DT stages within the presented model. In contrast, for studies that did not explicitly describe the model used, the identification was based on the outputs produced by DT and their relationship with SE.

The study’s findings reveal that most DT applications are integrated with RE activities (30 papers), primarily gathering detailed information about understanding user needs and expectations. The second most common application of DT, found in 6 papers, is in team development, aimed at equipping teams with the skills necessary to perform their roles as software developers. The distribution of DT integration stages with SE is illustrated in Table 6.

Table 6. DT integration in SE stage

Stage in SE	Study (Paper)	Percentage (%)	Integration strategy
Product evaluation	2	4.65%	Infused DT
RE activities	30	69.77%	Upfront DT
Risk/security analysis	1	2.33%	Infused DT
Team development	6	13.95%	Considered as Upfront DT
Whole process	4	9.30%	Continuous DT
Total	43	100.00%	

The integration of DT with software development, as outlined in Table 6, can be mapped based on the integration strategies proposed by Hehn *et al.* [14] and utilized by Parizi *et al.* [13]. These studies introduce the following terms: 1) Upfront DT, where DT is applied at the early stages of software development to understand customer needs and identify necessary software features; 2) infused DT, where DT is used as a tool to support existing RE activities; and 3) continuous DT, where DT is employed throughout the entire software development process. Based on this framework, the selected papers were mapped according to their respective integration strategies, as shown in Table 6 (integration strategy). This mapping provides a clear overview of how DT is integrated with software development in various research contexts, highlighting the different approaches taken to incorporate DT into the SE stages.

Product evaluation and risk/security analysis are categorized under infused DT because DT activities are used as tools to conduct evaluations and security analyses. In contrast, team development is considered upfront DT since DT is applied before the initial processes of SE, indicating that DT is utilized at the very start of the software development process. This approach ensures that the foundational aspects of the project, such as understanding user needs and defining necessary features, are well-established before moving forward. The research by Parizi *et al.* [13] supports this classification, asserting that upfront DT is the most commonly employed strategy in previous studies. This finding aligns with the current research results, confirming that upfront DT is also the most frequently used strategy in integrating SE with DT in education and training. This consistency highlights a prevalent preference for beginning software development with DT to ensure a comprehensive understanding of user needs and expectations from the outset, thereby enhancing the effectiveness of the development process.

3.3.4. RQ4 which research shows the DT and SE integration model in their works?

Not all studies explicitly describe the DT-SE integration model in their report. Only two of the 43 papers reviewed present an integration model. Alhazmi and Huang [43] introduce the integration of DT with the Scrum framework, utilizing an Upfront strategy where DT is applied at the beginning of the process to manage RE activities, resulting in outputs such as user stories and product backlogs. This model involves the incorporation of DT into Scrum iterations to accommodate adjustments in RE when a sprint has been completed. A visual representation of this model can be seen in Figure 6.

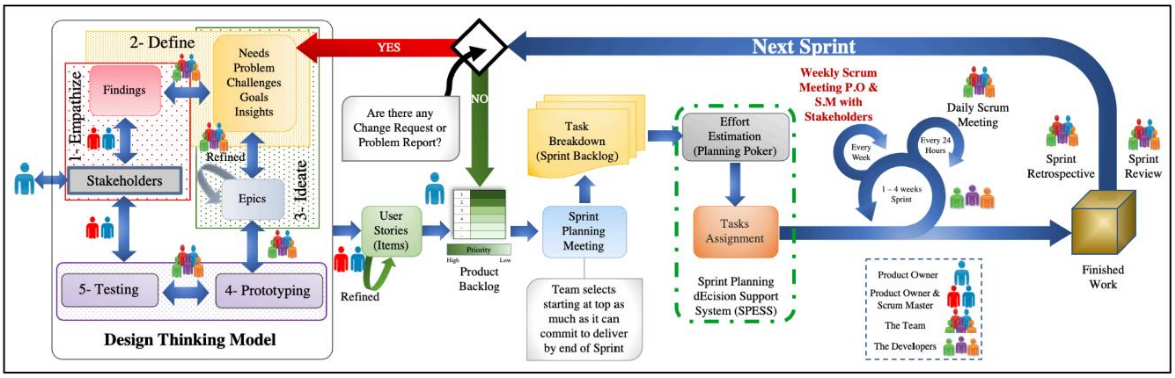

Figure 6. DT and scrum integration model [43]

This study employs the d.school DT model. The investigation shows that discussions on this integration DT model validation still need to be published, leaving its feasibility unconfirmed. Currently, the model's practicality and effectiveness need empirical evidence. Further research is necessary to establish its applicability. This presents an opportunity for future research to implement, test, or modify the application of the d.school model in these contexts. The study informs that several considerations should be considered when implementing this model. These include the complexity of the project scope, the experience level of the development team, and the clarity of the client's requirements. Addressing these factors is crucial for successfully applying the d.school model in educational and industrial environments.

Further, a study [6] developed an integrated model combining co-creation, DT, and ASD, specifically targeting digital banking innovation. The model is designed to present the roles of each approach, ensuring a structured and coherent process throughout. The implementation begins with the co-creation phase, where requirements are gathered, ensuring that diverse perspectives and insights are incorporated into the project's foundation. This collaborative approach allows identifying essential requirements and challenges from multiple sources, enriching the initial input. The DT phase generates ideas and creative solutions to address the identified needs. During this stage, outputs such as product canvases, user stories, and other design tools are produced, which serve as concrete guides for the development process. The development phase is then carried out using the ASD methodology (Scrum). The requirements gathered earlier are refined into a product backlog managed through Scrum's iterative and incremental development practices. This ensures that the product evolves through continuous feedback and adaptation, aligning with agile principles. The development process follows a structured approach where the product backlog items are prioritized and completed in cycles. Once the development phase is complete, the final product is ready for implementation, having undergone multiple iterations to ensure its alignment with user needs and project goals. A detailed depiction of this integrated model is presented in Figure 7.

Like the previous model, the developed model has yet to be officially published regarding its outputs. There is no evaluation of the model that can serve as a reference for further development. Additionally, this model only applies DT during the ideation phase, while the empathize and define phases are replaced by co-creation. Therefore, there are still opportunities for the further development, implementation, and evaluation of this model.

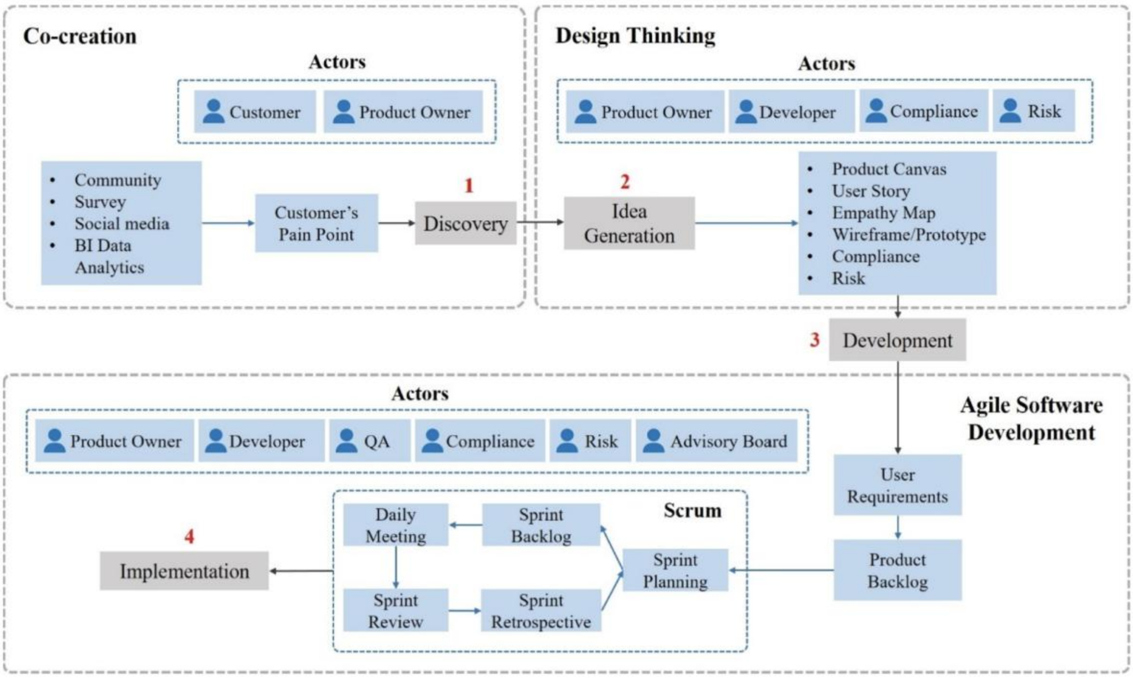

Figure 7. DT in the digital banking development process [6]

3.4. Discussion and limitation

This study reviewed 593 papers on DT integration with SE in education and training, subsequently selecting 43 key references. The Stanford d.school model was the most commonly applied (25 papers), followed by Google Design Sprint, Double Diamond, and IDEO. As a result, the most common issues observed in these works are categorized according to challenges, that were described in section 3.3.2. The challenges were categorized into team, process and management, complexity, and culture. Among the four categories, team challenges are the primary concern in implementing SE with DT, a trend consistent with DT research in a broader context. While solutions like tool development, frameworks, and evaluation methods exist, opportunities remain, particularly with generative AI, which is currently trending [46]. DT is mainly applied in RE [14], [59] due to its user-centered nature; however, its use in product evaluation and security analysis presents new research possibilities Because, at that stage, client/user involvement is essential in implementing it. Notably, only two studies explicitly described DT-SE integration models-one applied DT throughout the RE process [43], while the other focused on ideation [6]. Both generated backlogs from user stories and used Scrum.

Aside from this, there are notable differences and similarities that this study has discovered, compared to earlier studies conducted in the industry. On one hand, this study revealed that the d.school model from Stanford is the most widely adopted DT model, aligning with the findings by Parizi *et al.* [13] yet contrasting with the earlier findings by Russo and Pereira [12], which identified ISO as the prevailing model in the industry from 2008 to 2017 (see section 3.3.1). This transition indicates a shift within the industry from ISO to the d.school model over the past five years, a trend also mirrored in SE in education and training. The primary challenge highlighted is related to team dynamics, encompassing understanding, communication, and knowledge barriers, while the industry grapples with integration challenges and model selection, as noted by Russo and Pereira [12]. Furthermore, Parizi *et al.* [13] point out a deficiency in DT knowledge among teams, whereas in education, the emphasis lies more on developing soft skills. On the other hand, both this study and the previous findings by Parizi *et al.* [13] found that DT is mainly incorporated at the RE stage in SE (Upfront DT), whereas Russo and Pereira [12] noted a more extensive integration throughout SE. Overall, the findings of this study closely resonate with those of Parizi *et al.* [13] regarding model adoption, the challenges faced, and the stages of integration. This consistency is understandable given the differing timeframes examined: this study focuses on current developments from 2020 to 2024, while the work by Parizi *et al.* [13] considered 2010 to 2021, and Russo and Pereira [12] covered 2008 to 2017. Finally, one unexpected finding is the limited number of studies that explicitly visualize DT-SE integration models, suggesting a gap in structured frameworks for implementation. On the other hand, these insights suggest that SE in education and training aligns with industry trends, emphasizing both technical knowledge and soft skills, at least up to 2021.

This study comprehensively demonstrates that the integration of DT and SE in education and training aligns with its implementation in the industry regarding DT models, challenges, integration stage, and model. Although the challenges differ slightly, this is reasonable, as the primary focus of education and training is skills development. Furthermore, these findings contribute to educators and practitioners figuring out how to integrate DT and SE in education. Based on these findings, they could use or develop an integration model useful in education and training settings by applying these insights to shape curricula that emphasize user-centered design and collaboration. However, this study has limitations, such as excluding grey literature in the DT-SE integration model. Some models may not be published in peer-reviewed papers, meaning reports or technical documents discussing these topics were omitted, which could result in an incomplete capture of implementation outcomes. Furthermore, the potential future research, such as applying alternative DT models, proposing solutions to DT and SE integration challenges, especially in student team issues, exploring DT and SE integration beyond RE, and implementing and testing integration models developed in previous studies in education and training.

4. CONCLUSION

This study successfully identifies and analyzes relevant research on integrating DT and SE in education and training. Findings indicate that the d.school model, developed by Stanford, is the most widely adopted DT approach. However, alternative models such as Google Design Sprint, Double Diamond, and IDEO play significant roles. While the d.school model dominates, there is potential for alternative DT models, presenting opportunities for further research and application. Additionally, a comparison with previous studies reveals a shift in industry trends, with ISO being the dominant model until 2017, followed by the increasing adoption of d.school in both industry and education. However, while SE in education aligns with industry trends, model adoption in education is primarily driven by researchers and educators rather than direct industry adoption.

The study categorizes key challenges into four areas: team, process and management, complexity, and culture, with team-related issues being the most critical. A notable finding is that team challenges, such as understanding, communication, and knowledge barriers, are more prominent in education, whereas industry struggles more with integration processes and model selection. This is consistent with previous studies, which identified DT knowledge gaps in teams as a significant challenge. Existing solutions, including tools, frameworks, and evaluation methods, have shown progress, but only two studies explicitly define DT-SE integration models, both of which apply DT early in SE processes. The study also highlights the potential role of emerging technologies, such as generative AI, in addressing these challenges.

Finally, this study acknowledges the limitation of excluding grey literature, which may have resulted in an incomplete representation of available knowledge. Future research should consider a broader range of sources to develop structured integration frameworks, address team-related challenges, and expand DT applications beyond RE. Additionally, as SE in education continues to evolve, further exploration of how education can better adopt and adapt industry-based DT integration models is necessary to bridge the gap between academic research and industry practices. These findings also support educators and practitioners in selecting or developing integration models tailored for educational settings, helping shape curricula that emphasize user-centered design and collaboration.

FUNDING INFORMATION

This research was funded through a doctoral scholarship scheme for employees provided by the Yayasan Politeknik Chevron Riau (YPCR) in 2023.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Muhammad Ihsan Zul	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓
Suhaila Mohd. Yasin		✓		✓	✓			✓		✓		✓	✓	
Dadang Syarif		✓		✓	✓		✓			✓		✓		✓
Sihabudin Sahid														

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors state that they have no acknowledged financial conflicts of interest or personal relationships that could potentially influence the research discussed in this article. The authors indicate that there are no conflicts of interest.

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author, [Muhammad Ihsan Zul]. The data, which contains information that could compromise the privacy of research participants, is not publicly available due to certain restrictions.

REFERENCES

[1] J. Auernhammer and B. Roth, "The origin and evolution of Stanford University's design thinking: from product design to design thinking in innovation management," *Journal of Product Innovation Management*, vol. 38, no. 6, pp. 623–644, Nov. 2021, doi: 10.1111/jpim.12594.

[2] C. Meinel, L. Leifer, and H. Plattner, "Design thinking. understand – improve – apply. elektronische daten," 2011. http://search.ebscohost.com/login.aspx?direct=true&db=cat00327a&AN=stgal.000530013&lang=de&site=eds-live&authtype=ip,uid%5Cnhttp://dx.doi.org/10.1007/978-3-642-13757-0.

[3] J. Hehn and D. Mendez, "Combining design thinking and software requirements engineering to create human-centered software-intensive systems," in *Design Thinking for Software Engineering*, 2022, pp. 11–60.

[4] T. Gallanis, "An introduction to design thinking and an application to the challenges of frail, older adults," in *Leveraging Data Science for Global Health*, Cham: Springer International Publishing, 2020, pp. 17–33.

[5] N. Z. Mohamad, M. Chundau, and F. Y. Z. Yang, "Requirement analysis and problem finding using design thinking concepts in students' information system projects," in *2022 Applied Informatics International Conference (AiIC)*, May 2022, pp. 88–92, doi: 10.1109/AiIC54368.2022.9914584.

[6] E. Indriasari, H. Prabowo, F. L. Gaol, and B. Purwandari, "Adoption of design thinking, agile software development and co-creation: a qualitative study towards digital banking innovation success," *International Journal of Emerging Technology and Advanced Engineering*, vol. 12, no. 1, pp. 111–128, 2022, doi: 10.46338/IJETAE0122_11.

[7] F. Dobrigkeit, D. De Paula, and N. Carroll, "InnoDev workshop: a one day introduction to combining design thinking, lean startup and agile software development," *2020 IEEE 32nd Conference on Software Engineering Education and Training, CSEE and T 2020*, pp. 189–198, 2020, doi: 10.1109/CSEET49119.2020.9206184.

[8] L. Jia, N. A. Jalaludin, and M. S. Rasul, "Design thinking and project-based learning (DT-PBL): a review of the literature," *International Journal of Learning, Teaching and Educational Research*, vol. 22, no. 8, pp. 376–390, 2023, doi: 10.26803/ijlter.22.8.20.

[9] S. C. Noh and A. M. A. Karim, "Design thinking mindset to enhance education 4.0 competitiveness in Malaysia," *International Journal of Evaluation and Research in Education*, vol. 10, no. 2, pp. 494–501, 2021, doi: 10.11591/ijere.v10i2.20988.

[10] A. E. Lebid and N. A. Shevchenko, "Cultivation of the skills of design thinking via the project-based method as a component of the dual model of learning," *European Journal of Contemporary Education*, vol. 9, no. 3, pp. 572–583, 2020, doi: 10.13187/ejced.2020.3.572.

[11] V. G. Ferreira and E. D. Canedo, "A design sprint based model for user experience concern in project-based learning software development," *Proceedings - Frontiers in Education Conference, FIE*, vol. 2020-October, 2020, doi: 10.1109/FIE44824.2020.9274214.

[12] R. de F. S. M. Russo and J. C. Pereira, "Design thinking integrated in agile software development: a systematic literature review," *Procedia Computer Science*, vol. 138, pp. 775–782, 2018.

[13] R. Parizi, M. Prestes, S. Marczak, and T. Conte, "How has design thinking being used and integrated into software development activities? A systematic mapping," *Journal of Systems and Software*, vol. 187, 2022, doi: 10.1016/j.jss.2022.111217.

[14] J. Hehn, D. Mendez, F. Uebernickel, W. Brenner, and M. Broy, "On integrating design thinking for human-centered requirements engineering," *IEEE Software*, vol. 37, no. 2, pp. 25–31, Mar. 2020, doi: 10.1109/MS.2019.2957715.

[15] L. Corral and I. Fronza, "Design thinking and agile practices for software engineering," in *Proceedings of the 19th Annual SIG Conference on Information Technology Education*, Sep. 2018, pp. 26–31, doi: 10.1145/3241815.3241864.

[16] B. Kitchenham, "Procedures for performing systematic reviews," *Keele University Technical Report*, vol. 33, no. 2004, pp. 1–26, 2014, [Online]. Available: https://www.researchgate.net/publication/228756057.

[17] B. Kitchenham and P. Brereton, "A systematic review of systematic review process research in software engineering," *Information and Software Technology*, vol. 55, no. 12, pp. 2049–2075, Dec. 2013, doi: 10.1016/j.infsof.2013.07.010.

[18] M. Sanchez-Gordon, R. Mendoza-Gonzalez, and R. Colomo-Palacios, "Design thinking in practice," *IT Professional*, vol. 23, no. 4, pp. 95–100, Jul. 2021, doi: 10.1109/MITP.2020.2993113.

[19] A. Johnson, "Design thinking with SAP," *SAP Press*, 2018. .

[20] R. Parizi, M. Moreira, I. Couto, S. Marczak, and T. Conte, "A design thinking techniques recommendation tool: an initial and on-going proposal," in *19th Brazilian Symposium on Software Quality*, Dec. 2020, pp. 1–6, doi: 10.1145/3439961.3439997.

[21] Q. V. Liao, H. Subramonyam, J. Wang, and J. W. Vaughan, "Designerly understanding: information needs for model transparency to support design ideation for AI-powered user experience," in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, Apr. 2023, pp. 1–21, doi: 10.1145/3544548.3580652.

[22] R. Jolak, A. Wortmann, G. Liebel, E. Umuhoza, and M. R. V. Chaudron, "The design thinking of co-located vs. distributed software developers," in *Proceedings of the 15th International Conference on Global Software Engineering*, Jun. 2020, pp. 106–116, doi: 10.1145/3372787.3390438.

[23] Y. Masuda, A. Zimmermann, D. S. Shepard, R. Schmidt, and S. Shirasaka, "An adaptive enterprise architecture design for a digital healthcare platform : toward digitized society – Industry 4.0, Society 5.0," in *2021 IEEE 25th International Enterprise Distributed Object Computing Workshop (EDOCW)*, Oct. 2021, pp. 138–146, doi: 10.1109/EDOCW52865.2021.00043.

[24] A. W. Ramadhan, D. S. Kusumo, and M. K. Sabariah, "Analysis of development performance and product quality with scrum design thinking development (case study: e-recruitment)," in *2022 1st International Conference on Software Engineering and Information Technology (ICoSEIT)*, Nov. 2022, pp. 228–232, doi: 10.1109/ICoSEIT55604.2022.10030080.

[25] R. Chatley, T. Field, M. Wheelhouse, C. Runcie, C. Grinyer, and N. De Leon, "Designing for real people: teaching agility through user-centric service design," in *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, May 2023, pp. 11–22, doi: 10.1109/ICSE-SEET58685.2023.00007.

[26] S. Jantunen, A. Pesola, P. Janhunen, T. Reijonen, and V. Kuuluvainen, "Developing wellbeing service effectiveness platform : some RE considerations," in *2020 IEEE First International Workshop on Requirements Engineering for Well-Being, Aging, and Health (REWBAH)*, Aug. 2020, pp. 19–24, doi: 10.1109/REWBAH51211.2020.00009.

[27] E. Indriasari, H. Prabowo, F. L. Gaol, and B. Purwandari, "The adoption of design thinking, agile software development and co-creation concepts a case study of digital banking innovation," in *2021 International Conference on Platform Technology and Service (PlatCon)*, Aug. 2021, pp. 1–6, doi: 10.1109/PlatCon53246.2021.9680763.

[28] M. Zorzetti, I. Signoretti, L. Salerno, S. Marczak, and R. Bastos, "Improving agile software development using user-centered design and lean startup," *Information and Software Technology*, vol. 141, p. 106718, Jan. 2022, doi: 10.1016/j.infsof.2021.106718.

[29] W. Hussain *et al.*, "Human values in software engineering: contrasting case studies of practice," *IEEE Transactions on Software Engineering*, vol. 48, no. 5, pp. 1818–1833, May 2022, doi: 10.1109/TSE.2020.3038802.

[30] F. Dobrigkeit, C. Matthies, P. Pajak, and R. Teusner, "Cherry picking - agile software development teams applying design thinking tools," in *Lecture Notes in Business Information Processing*, vol. 426, 2021, pp. 201–206.

[31] M. Prestes, R. Parizi, S. Marczak, and T. Conte, "On the use of design thinking: a survey of the Brazilian agile software development community," in *Lecture Notes in Business Information Processing*, vol. 383 LNBIP, 2020, pp. 73–86.

[32] C. Arora, L. Tubino, A. Cain, K. Lee, and V. Malhotra, "Persona-based assessment of software engineering student research projects: an experience report," in *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, May 2023, pp. 198–209, doi: 10.1109/ICSE-SEET58685.2023.00025.

[33] S. Gottschalk, E. Yigitbas, A. P. Nowosad, and G. Engels, "Towards software support for situation-specific cross-organizational design thinking processes," in *Proceedings of the 5th International Workshop on Software-intensive Business: Towards Sustainable Software Business*, May 2022, pp. 1–8, doi: 10.1145/3524614.3528624.

[34] A. Hinderks, F. J. Domínguez Mayo, J. Thomaschewski, and M. J. Escalona, "Approaches to manage the user experience process in Agile software development: a systematic literature review," *Information and Software Technology*, vol. 150, p. 106957, Oct. 2022, doi: 10.1016/j.infsof.2022.106957.

[35] S. Trieflinger, D. Lang, S. Spies, and J. Münch, "The discovery effort worthiness index: how much product discovery should you do and how can this be integrated into delivery?," *Information and Software Technology*, vol. 157, p. 107167, May 2023, doi: 10.1016/j.infsof.2023.107167.
[36] D. Karagiannis, R. A. Buchmann, and W. Utz, "The OMiLAB digital innovation environment: agile conceptual models to bridge business value with digital and physical twins for product-service systems development," *Computers in Industry*, vol. 138, p. 103631, Jun. 2022, doi: 10.1016/j.compind.2022.103631.
[37] S. De, "A novel perspective to threat modelling using design thinking and agile principles," in *2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, Nov. 2020, pp. 31–35, doi: 10.1109/PDGC50313.2020.9315844.
[38] D. Zurita-Gaibor, M. Escobar-Sánchez, and X. López-Chico, "Application of 'Design Thinking' in the development of virtual platforms with gamified elements," in *2022 Third International Conference on Information Systems and Software Technologies (ICI2ST)*, Nov. 2022, pp. 122–129, doi: 10.1109/ICI2ST57350.2022.00025.
[39] R. Adinegoro, S. Suakanto, H. Fakhrurroja, and M. Hardiyanti, "Comparison of UI/UX development using design thinking vs lean UX : a comparative study," *Proceedings - 2023 3rd International Conference on Electronic and Electrical Engineering and Intelligent System: Responsible Technology for Sustainable Humanity, ICE3IS 2023*, pp. 147–152, 2023, doi: 10.1109/ICE3IS59323.2023.10335225.
[40] M. Chouki, B. B. de Mozota, A. Kallmuenzer, S. Kraus, and M. Dabic, "Design thinking and agility in digital production: the key role of user experience design," *IEEE transactions on on Engineering Management*, 2021, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9539885/.
[41] W. Hussain *et al.*, "How can human values be addressed in agilemethods a case study on SAFe," *IEEE Transactions on Software Engineering*, vol. 48, no. 12, pp. 1–1, 2022, doi: 10.1109/TSE.2022.3140230.
[42] M. Sánchez-Gordón, R. Colomo-Palacios, A. Sánchez, and S. Sanchez-Gordon, "Integrating approaches in software development: a case analysis in a small software company," in *Communications in Computer and Information Science*, vol. 1251 CCIS, 2020, pp. 95–106.
[43] A. Alhazmi and S. Huang, "Integrating design thinking into scrum framework in the context of requirements engineering management," in *Proceedings of the 2020 3rd International Conference on Computer Science and Software Engineering*, May 2020, pp. 33–45, doi: 10.1145/3403746.3403902.
[44] J. Vilela and C. Silva, "An experience report on the use of problem-based learning and design thinking in a requirements engineering postgraduate course," in *Proceedings of the XXXVII Brazilian Symposium on Software Engineering*, Sep. 2023, pp. 432–441, doi: 10.1145/3613372.3614188.
[45] H. Gunatilake, J. Grundy, R. Hoda, and I. Mueller, "Enablers and barriers of empathy in software developer and user interactions: a mixed methods case study," *ACM Transactions on Software Engineering and Methodology*, vol. 33, no. 4, pp. 1–41, May 2024, doi: 10.1145/3641849.
[46] A. R. Asadi, "LLMs in design thinking: autoethnographic insights and design implications," in *2023 The 5th World Symposium on Software Engineering (WSSE)*, Sep. 2023, pp. 55–60, doi: 10.1145/3631991.3631999.
[47] R. C. Fuller and P. Kruchten, "Blurring boundaries: toward the collective empathic understanding of product requirements," *Information and Software Technology*, vol. 140, p. 106670, Dec. 2021, doi: 10.1016/j.infsof.2021.106670.
[48] F. H. Lermen, P. K. de Moura, V. B. Bertoni, P. Graciano, and G. L. Tortorella, "Does maturity level influence the use of Agile UX methods by digital startups? Evaluating design thinking, lean startup, and lean user experience," *Information and Software Technology*, vol. 154, p. 107107, Feb. 2023, doi: 10.1016/j.infsof.2022.107107.
[49] A. T. S. Calazans, A. J. Cerqueira, and E. D. Canedo, "Empathy and criativity in privacy requirements elicitation: systematic literature review," 2020, doi: 10.29327/1298730.23-17.
[50] M. S. Mirza and S. Datta, "Developing software using agile and design thinking framework," *Proceedings - 2020 International Conference on Computational Science and Computational Intelligence, CSCI 2020*, pp. 1819–1823, 2020, doi: 10.1109/CSCI51800.2020.00335.
[51] C. Péraire, "Learning to write user stories with the 4C model: context, card, conversation, and confirmation," in *2023 IEEE/ACM 5th International Workshop on Software Engineering Education for the Next Generation (SEENG)*, May 2023, pp. 33–36, doi: 10.1109/SEENG59157.2023.00011.
[52] S. De, "Minimal satisfiable product: a novel concept in agile project management for better adherence to customer requirements," in *2021 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)*, Oct. 2021, pp. 1–6, doi: 10.1109/SMARTGENCON51891.2021.9645781.
[53] A. Poth and A. Riel, "Quality requirements elicitation by ideation of product quality risks with design thinking," in *2020 IEEE 28th International Requirements Engineering Conference (RE)*, Aug. 2020, vol. 2020-Augus, pp. 238–249, doi: 10.1109/RE48521.2020.00034.
[54] R. Sharma and J. N. Singh, "Systematic literature review and comparative studies on human centred software development," *Proceedings - 2022 4th International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2022*, pp. 2239–2246, 2022, doi: 10.1109/ICAC3N56670.2022.10074154.
[55] P. Vesikivi, M. Bauters, and J. Holvikivi, "Agile methodologies in learning with design thinking," in *Lecture Notes in Educational Technology*, 2021, pp. 75–90.
[56] A. Matthiesen, G. Majgaard, and L. J. Larsen, "Overcoming social anxiety: how virtual reality and game-based elements are revolutionizing patient therapy.," *European Conference on Games Based Learning*, vol. 17, no. 1, pp. 782–788, Sep. 2023, doi: 10.34190/ecgbl.17.1.1734.
[57] M. Videnovik, V. Trajkovik, L. V. Kiønig, and T. Vold, "Increasing quality of learning experience using augmented reality educational games," *Multimedia Tools and Applications*, vol. 79, no. 33–34, pp. 23861–23885, Sep. 2020, doi: 10.1007/s11042-020-09046-7.
[58] M. Schmidt *et al.*, "Learning experience design of an mHealth self-management intervention for adolescents with type 1 diabetes," *Educational technology research and development*, vol. 70, no. 6, pp. 2171–2209, Dec. 2022, doi: 10.1007/s11423-022-10160-6.
[59] A. Husaria and S. Guerreiro, "Requirement engineering and the role of design thinking," in *Proceedings of the 22nd International Conference on Enterprise Information Systems*, 2020, vol. 2, pp. 353–359, doi: 10.5220/0009489303530359.

BIOGRAPHIES OF AUTHORS

Muhammad Ihsan Zul is a lecturer at Politeknik Caltex Riau, Indonesia. He holds a master's degree in electrical engineering and information technology from Gadjah Mada University, Indonesia. Currently, he is a Ph.D. student at the Faculty of Computer Science and Information Technology University Tun Hussein Onn Malaysia (UTHM). His dissertation topic is software engineering, focusing on requirement engineering and design thinking in education. He can be contacted at email: ihsan@pcr.ac.id.

Suhaila Mohd. Yasin received a B.Sc. degree in computing and an M.Sc. degree in computer science from Universiti Teknologi Malaysia, Skudai, and a Ph.D. degree in computer science from The University of Queensland, Australia, in 2020. She is currently a senior lecturer with the Department of Software Engineering and the Software Testing Focus Group leader and principal researcher at Universiti Tun Hussein Onn Malaysia (UTHM). As a rising researcher, she currently supervises postgraduate students and has authored or co-authored articles in the fields of software engineering, software modeling, development, and testing. She can be contacted at email: ysuhaila@uthm.edu.my.

Dadang Syarif Sihabudin Sahid received a bachelor's degree in mathematics from Bandung Institute of Technology, Indonesia, in 1999. The M.Sc. degree in Information Technology was earned at the University Teknologi Malaysia in 2009. Doctoral holder from Universitas Gadjah Mada, Indonesia, in 2018. He has been a lecturer at the Information Technology Department of Politeknik Caltex Riau since July 1st, 2000. As a lecturer, he carries out three main activities: teaching, research, and community service. Currently, he teaches in the fields of information systems, application and systems development, IT project management, information technology concepts, and IT strategic planning. The areas of research that are focused on include IT project management, soft computing, context aware computing, and IT planning. He received many research grants, such as collaboration grants and higher education ministry grants, including grants from abroad. He also has experience as deputy director and director at the Politeknik Caltex Riau. He can be contacted at email: dadang@pcr.ac.id.