# It Takes Two to Tango: Design Thinking and Design Patterns for Better System Development

Ernestine Dickhaut[1], Andreas Janson[2] and Jan Marco Leimeister[1,2]

[1] Information Systems, University of Kassel, Kassel, Germany
[ernestine.dickhaut, leimeister]@uni-kassel.de
[2] Institute of Information Management, University of St.Gallen, St. Gallen, Switzerland
{andreas.janson, janmarco.leimeister}@unisg.ch

**Abstract.**
Design Thinking has become a well-established approach to solving wicked problems through creative and conceptual solutions. Thus, De-sign Thinking approaches usually end with finding novel solutions but do not offer support in the practical implementation of these solutions. To bridge the gap between finding creative solutions and implementing them into suitable end products, we see De-sign Patterns as a useful ap-proach to combine the advantages of both sides—the human-centered Design Thinking approach and the practical implementation in system development. Design Patterns offer proven solutions to recurring prob-lems and thus provide design knowledge to solve complex design prob-lems. In this contribution, we demonstrate how Design Patterns act as a complement to Design Thinking by using the example of designing a law-ful smart personal learning assistant. We use Design Thinking to extract the design solution of the lawful learning assistant and develop it with the help of Design Patterns. We demonstrate the use of Design Patterns by using the deletion routine Design Pattern as an example of how a De-sign Pattern can be used for lawful design in addition to their known fields of application in system development. The evaluation results show that the combination of Design Thinking and Design Patterns lead to an approach that not only identifies novel, complex solutions but also support their practical implementation.

**Keywords:** Design Pattern, Design Thinking, Smart Personal Assistants, Lawful System Development

## 1 Introduction

Smart personal assistants, such as Alexa and Siri, are becoming increasingly popular [1]. They are not only used for private purposes in everyday life, but also increasingly in learning contexts [2–4]. The use of smart personal learning assistants (SPLA) in university teaching differs from their use in private households. Both data protection regulations and the didactic added value of the system must meet the requirements of the university. Thus, many different disciplines, such as didactics, psychology, law, or computer science, are involved in the development of an SPLA for use in teaching [2].

This results in multidimensional requirements for the SPLA, which must be combined into one system.

For the analysis of the complex and conflicting requirements, as well as the development of a design solution, design thinking offers a useful framework. Design thinking has become a well-established approach to developing products, services, processes, and business models by improving problem-solving by people from different disciplines working together to either solve a problem or develop new ideas. The approach is primarily used to solve so-called wicked problems, which are considered to be particularly complex and inconsistent in finding a solution. These problems require creative and conceptual solutions, which can lead to challenges in prototypical or final product design. Thus, design thinking end with finding a novel solution that does not support developer to implement the discovered solution.

Design patterns may be a valuable tool to support the implementation of those novel design solutions. Design patterns provide proven solutions for recurring problems and are mainly established in the field of system development and support developers. We see both theoretical and practical potential in combining both approaches to successfully develop novel artifacts in a sustainable way. The goal of our paper is to use the advantages of the user-centered design thinking approach and provide a bridge to system development through the use of design patterns that support developers in the implementation of design solutions and is based on the following research question (RQ):

**RQ**: How can we combine the advantages of the user-centered Design Thinking approach with Design Patterns to design useful, novel IT artifacts?

To answer our research question, we use the example of developing a lawful SPLA. For the development of the SPLA, we use design thinking to extract creative solutions of possible SPLAs in an interdisciplinary team of legal experts, developers, and dialecticians. We then use design pattern to implement these solutions in a prototype, which we then evaluate with users and legal experts using the law simulation study.

## 2 Related Work

### 2.1 Design Thinking

Design thinking is a structured problem-solving approach to solve real-world problems [5]. In recent years, different interpretations of design thinking have be-come established in both research and practice. Depending on the ap-proach and goal, different methods and artifacts are pursued and devel-oped. The approaches aim to improve problem-solving by having people from different disciplines work together to either solve a problem or to develop new ideas. Through novel workshop settings and usually a new environment in the workshops, the creativity of the participants is en-couraged. Design Thinking focuses on the needs and motivations of the user and can be seen as a human-centered approach. In system devel-opment, we pretend too often that requirements somehow just exist and that they simply need to be elicited and documented, thus missing the true potential of fully solving the problem in a human-centric man-ner. Design Thinking addresses this problem and offers various tools for practical

application. Design thinking can be seen as a process perspective, which can be used to structure the procedure from identifying the problem to finding suitable solutions. Thus, many variants have become established in research and practice. In our paper, we focus on the following phases: understanding, observing, defining the problem, ideating, prototyping, and testing (see Figure 1).
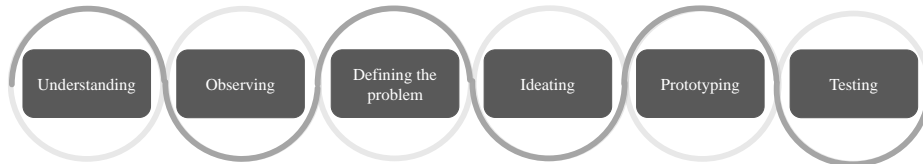


**Fig. 1.** Design Thinking Process

## 2.2 Design Patterns

Design patterns originally came from the field of architecture and were established in the work of Alexander et al. [6], in which they support architects in the designing of houses and the planning of cities. In design, architects often face recurring problems to which proven solutions can be applied. Therefore, design patterns codify proven solutions for recurring problems and make them usable for the future.

Thus, the use of patterns has become established in various disciplines. In Human-Computer Interaction (HCI), patterns have already been proven in many studies to teach design principles and design concepts [7–9]. In system development, patterns were first established through the Gang of Four (GoF) [10]. In addition to the previously used application areas, patterns can be used to enable a broad understanding of periphery disciplines [11].

In system development, design patterns represent abstract and thus generally applicable and reusable solutions for recurring problems. Design patterns codify "best practices" and make them usable for the future. For this purpose, a design pattern offers a type of "template" that follows the same structure for all patterns. In terms of content, design patterns do not present innovative solutions and do not reinvent the wheel but rather build on proven solutions. One design pattern offers solutions for many different problems. Thus, design patterns can counteract one of the problems identified by vom Brocke et al. [12] in the reuse of design knowledge by codifying said knowledge into an abstract form in design patterns.

According to Nonaka [13], design knowledge is represented as implicit knowledge. Thus, the knowledge must be externalized to make it accessible to other people. This can be achieved either by talking about it or by writing it down and codifying it. Design patterns externalize implicit design knowledge by codifying it and capturing it in written form. According to vom Brocke et al. [12], design patterns are a component of design knowledge base. Patterns help to find suitable solutions for existing problems. By providing information about the context of the problem, a design pattern helps to understand the problem in detail and thus find the most suitable solution. In addition to the description of the problem, one core component of a pattern is the solution and a corresponding description of the solution procedure. In addition to the familiar form of

design pattern (for example, from system development), modified forms with new purposes have become established in recent years, for example, legal design pattern, which is considered in this paper.

## 3 Developing Smart Personal Learning Assistants Using Design-Thinking and Design Patterns

In the following, we show the development of the SPLA using design thinking and design patterns (Figure 2). First, we used design thinking to collaborate in an interdisciplinary team consisting of dialecticians, lawyers, and developers to extract design solutions for a lawful SPLA that includes innovative didactic learning concepts and thus conveys the learning content to the user. Second, to support the development of the SPLA, we used design patterns. The design patterns were used where design thinking ends. We used design patterns that had been developed in another project to design AI-based assistants in a lawful way. We classified the SPLA as a sub-group of AI-based assistants and thus provided developers with support for implementing creative design solutions in practice.
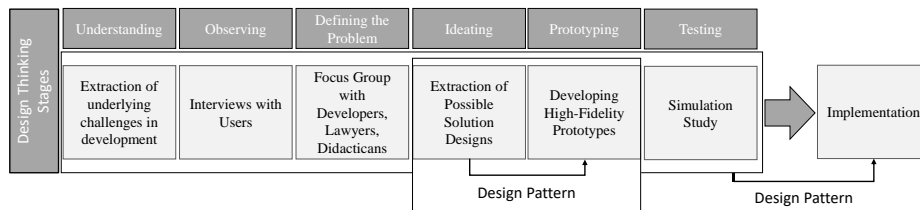
| Design Thinking Stages | Understanding | Observing | Defining the Problem | Ideating | Prototyping | Testing | |
|---|---|---|---|---|---|---|---|
| | Extraction of underlying challenges in development | Interviews with Users | Focus Group with Developers, Lawyers, Didacticans | Extraction of Possible Solution Designs | Developing High-Fidelity Prototypes | Simulation Study | Implementation |

Design Pattern      Design Pattern

**Fig. 2.** Design Thinking Stages

### 3.1 Finding the Design Solution using Design Thinking

The SPLA development project starts with Design Thinking to elaborate design solutions. Thus, an interdisciplinary team from three disciplines—law, didactics, and computer science—was involved in the development of the SPLA, as these three disciplines differ in their methodologies and their viewpoint and thus in their requirements for an SPLA. To involve all disciplines equally in the development and to develop a common design solution, we follow the structured procedure of Design Thinking. In the first phase, understanding, we first conducted a joint workshop of all in-volved persons. The goal of the workshop was to extract the challenges in the development of a lawful SPLA.

After enabling a shared understanding of the other disciplines within the project team, the focus was placed on the user of the SPLA. For this purpose, we conducted interviews with potential SPLA users, seven students (N = 7). The interviews allowed us to integrate the user-centered thoughts of Design Thinking into the development. The sub-ject of the discussion was, for example, the storage of user data. To en-able adaptive learning from a didactic perspective, the user must be uniquely identifiable,

and the progress must be stored and processed. But legal experts are critical of the idea, since this involves personal data that is particularly worthy of protection.

Based on the previous workshops and interviews, the project team col-laborated on possible design solutions. So also, to find a solution for the storage and processing of personal data. For this purpose, the Design Pattern "deletion routine" is used during the Design Thinking process, because deletion routines present a possible solution to the problem. Many different creative solutions came out of this. As usual for Design Think-ing, in this phase, solutions are also generated that cannot be im-plemented in practice. To avoid this, we used Design Patterns already at this stage. The Design Patterns sup-ported the project team in thinking about the practical implementation of the design solution at an early stage as well as support them to implement the design solution later.

In our specific project, we created three design solutions for an SPLA in higher ed-ucation. The next step was to implement one solution in an SPLA. For the transfer from the Design Thinking phase to the actual im-plementation, a team of three experienced developers was given both the design solution and the Design Pattern. The Design Pat-terns provid-ed proven solutions for the implementation of similar design problems.

### 3.2 Developing the Design Solution using Design Patterns

In the following, we describe how we developed the extracted design solutions from design thinking and implemented them with the support of design patterns in an SPLA. The used design patterns are a well-established approach in different projects in infor-mation system research (please see [14–16]). With the design pattern, the previously known use in system development was expanded by providing proven solutions to legal problems in system development for AI-based assistants. The design patterns provide proven solutions for the development of the lawful SPLAs and were developed and evaluated in the context of another study (for more details, please see [17]).

The SPLA was used in the context of a university course as a complementary prep-aration for exams, and, together with the user, it repeated the course material in a playful way. In the learning quiz, the SPLA asked the user a question and offered four possible answers, of which exactly one answer was correct. The used teaching material was based on the content of the course and was prepared together with the lecturer and other teachers of the course and implemented into the SPLA.

The design pattern, *deletion routine* (Figure 3), implied that the purpose of the data processing had to be specified precisely and differentiated. It also suggested a granular consent option for this purpose. The proposed solutions from the design pattern were implemented in the development of the SPLA, and personalization was initially deac-tivated in the default settings. This allowed the user to select exactly which data storage option they wanted. For this, the user received additional information, options that ex-plained exactly why the respective data storage was necessary, and what added value it had for achieving the goal. In addition, to ensure transparency with regard to all data processing operations, which was recommended in the design pattern, we designed the SPLA in such a way that users were adaptive informed about the data processing oper-ations. The user was provided with appropriate information, not only during the first use, but whenever relevant data processing took place. To balance transparency and

trouble-free use, we provided the option to select how often and how detailed they would like to be informed about data processing. In addition, as recommended in the design pattern, users had the option of viewing their digital self-disclosure about the storage of their data in the settings at any time. This supported the transparency of data storage.

| Deletion Routines | | Current Period of Development Process | |
|---|---|---|---|
| | | ☐ Patterns of interaction | ☒ Architectural patterns |
| | | ☐ Learning patterns | ☒ Data processing pattern |

| Goal | | |
|---|---|---|
| Erasure of personal data as soon as they are no longer necessary to achieve the purpose of processing. | | |

| Requirements | | |
|---|---|---|
| **Law** | | **Service quality** |
| • Differentiated uses <br> • Non-chain-ability <br> • Avoidance of personal data | • No complete user profile | • Motivation of the provider <br> • Learning through relevance assessments <br> • Secondary function <br> • Remember Me |

| Consequences | | Influences |
|---|---|---|
| **Law** <br> • Ownership of the user over his data <br> • Right to be forgotten | **Service quality** <br> • Analysis and categorisation of data according to those that are subject to deletion and retention obligations, timely deletion of the data in compliance with the law. | • Data minimisation <br> • Appropriation <br> • Protection of privacy and intimacy <br> • Functionality <br> • Configurability |

**Solution**

Integration of a deletion concept:
• Localisation of personal data, on systems personal data is stored
• Analysis of data with regard to retention and deletion obligations
• Determine deletion and retention periods, group data according to these periods
• Define deletion rules for the individual groups ordered by deadlines.
• Delete or anonymise data, it is important to delete all records in all software systems.
• Data of individuals must be retrievable and separately deletable

**Important Data Protection Regulations**
• Art. 5 para. 1 lit. b (Purpose limitation), lit. c (Data minimisation), lit. e (Storage limitation) (Here, where applicable, opening clauses such as Art. 6 para. 3 of the GDPR and member state regulations based on this in the BDSG, HDSIG and HHG must also be observed, especially for data processing by public bodies).
• Art. 17 of the GDPR (Right to erasure) (Member State regulations based on Art. 23 of the GDPR may need to be taken into account here).

| Confirmation of implementation of contents of design pattern | Date | Signature |
|---|---|---|

**Fig. 3.** Design Pattern "Deletion Routines" [15]

To meet the requirements of law on purpose limitation, the design pattern recommended a regular check, in which personal data was still required to achieve the purpose. Deletion routines were a solution approach for this. In our SPLA, we saw the possibility of performing such automated deletion routines, especially after the end of a semester.

## 4 Overall Evaluation of the Development Process and Legal Assessment

### 4.1 Evaluation of the Development Process

To the best of our knowledge, with this contribution, we are the first who combine the two approaches design thinking and design pattern, and demonstrate the approach with

a concrete development example. In the following section, we would like to look back at the procedure and evaluate it.

Design thinking provides a framework that enables interdisciplinary project teams to develop novel design solutions in a structured way. One of the biggest challenges at the beginning of the project was the different approaches and languages of the individual disciplines. Through the workshops in the first stages, the project team was able to broaden their own perspectives and understand the requirements of the other disciplines. This made it possible to find a common language within the project team. The structured framework supports the project team in the individual stages without limiting them to specific methods.

The combination of design thinking and design patterns lead to the development of runnable novel technologies. The challenge of combining complex and sometimes conflicting requirements in an SPLA was solved by design thinking. Innovative design solutions were extracted, which could then be implemented with the support of the design pattern. Usually, design thinking ends with the prototypical implementation and evaluation of a design solution. By using the design pattern we build a bridge to the final implementation and support the developer in the implementation of the solution.

## 4.2    Legal Assessment of the developed SPLA

We conducted a law simulation study which is a well-established evaluation method under lawyer [18] to evaluate the SPLA (Figure 4). The simulation study can generally be divided into two parts. In the first part, the system was examined in a practical evaluation with users for its usability, user experience, and potential application problems. The second part of the simulation study served to evaluate the legality of the SPLA. In this second part of the study, simulated court cases were conducted in which the system was evaluated with the participation of real judges and lawyers for its lawfulness.

To investigate the primary aim of the first part of the simulation study, the use of the SPLA in practice by real users was essential. For the evaluation, we offered an exam preparation course in a university basic economics course that complemented the lecture, in which students could participate voluntarily.

On three dates in a period of two weeks, the participants could study together with the SPLA for one hour for the exam and repeat the acquired knowledge from the course. The offer took place on a voluntary basis. We then used questionnaires to survey student satisfaction with the SPLA. In addition, we asked in a qualitative survey for suggestions on improving the SPLA in the future and increasing its added value for students and lecturers.

Based on the user evaluation, we conducted four simulated court cases (for further information on the evaluation, please see [14, 15, 19]. The simulation study was carried out before German courts according to German and European law. Overall, six legal experts participated in our law simulation study. Among them were two judges and four lawyers that conducted the four court cases. All participants had completed the second state examination in law and already had several years of professional experience as a lawyer or judge. One participant was female, the other five males. The oral hearings lasted 45 and 60 minutes. Each of the four trials involved a judge, a lawyer from the

defendant, a lawyer from the plaintiff, the plaintiff, and the defendant. For the plaintiffs, we recruited voluntary participants from the first part of the simulation study to present the process as realistically as possible. In all four cases, the defendant's side was represented by the university, which used the IT artifact in the lecture course.
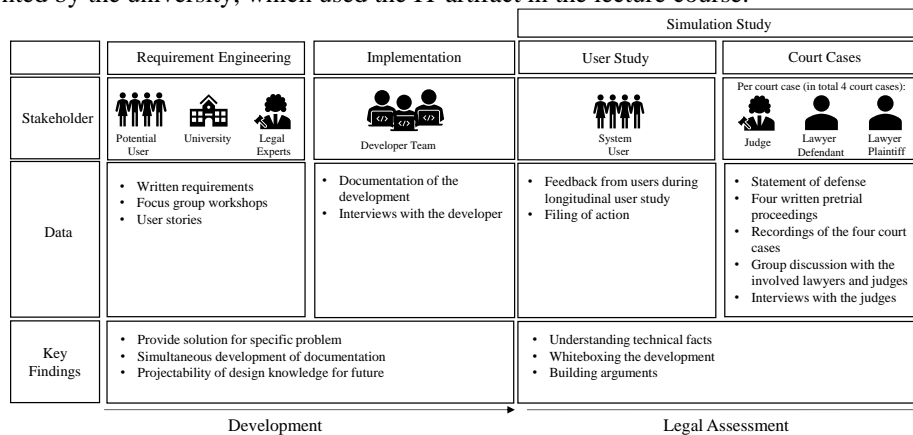
| | Requirement Engineering | Implementation | Simulation Study | |
| --- | --- | --- | --- | --- |
| | | | User Study | Court Cases |
| Stakeholder | Potential User · University · Legal Experts | Developer Team | System User | Per court case (in total 4 court cases): Judge · Lawyer Defendant · Lawyer Plaintiff |
| Data | • Written requirements<br>• Focus group workshops<br>• User stories | • Documentation of the development<br>• Interviews with the developer | • Feedback from users during longitudinal user study<br>• Filing of action | • Statement of defense<br>• Four written pretrial proceedings<br>• Recordings of the four court cases<br>• Group discussion with the involved lawyers and judges<br>• Interviews with the judges |
| Key Findings | • Provide solution for specific problem<br>• Simultaneous development of documentation<br>• Projectability of design knowledge for future | | • Understanding technical facts<br>• Whiteboxing the development<br>• Building arguments | |
| | Development → | | Legal Assessment → | |

**Fig. 4.** Simulation Study [14]

Before the oral hearings, written preliminary proceedings took place in which the plaintiff's lawyer set out the facts of the case and the reasons for the action and called on the defendant to refrain from using the IT artifact in university teaching. The reason for one of the four actions was the *collection of personal data beyond the purpose of processing*, as well as information about the duration and purpose of data storage. In a five-page statement of defense, the defendant's lawyer commented on the action. In the statement of defense, the lawyer referred to the *deletion routine* design pattern that was used in the development process of the SPLA.

Thereupon, the judge invited the participants to an oral hearing to dispute the action. In court, the judge first presented the facts of the case and discussed the reasons for the action. After the plaintiff's lawyer confirmed the facts of the case and set out the reason for the action in more detail, the two lawyers and the judge examined the facts. The reason for the action was the processing of personal data beyond the actual purpose of use, namely, the adaptation of the SPLA to the learner. The plaintiff participated in exam preparation with the SPLA last semester. The plaintiff claims to have achieved poor results in the exam preparation due to private problems. Nevertheless, she passed the exam. When the plaintiff applied for a job at the department a few months after the end of the semester and was rejected, the plaintiff saw a connection between the poor results in the exam preparation with the help of the SPLA and her not getting the job, so she claimed that the data and results of the SPLA were used beyond their intended purpose. Both parties then had the opportunity to present their side and the judge could get an impression of the situation.

According to the plaintiff, the defense lawyer came to state his case and referred in his statement to the implementation of regularly deleting routines at the end of the semester. To confirm this technically, he also referenced the design pattern, *deletion*

*routine*, used in the development of the SPLA. In the course of the hearing, it turned out that the responsible chair had already deleted the data at the time of application and that it was not used beyond its intended purpose, even during its use. The negotiations ended with the pronouncement of a judgment, which was in favor of the university. In addition to the four court cases and the written correspondence, we interviewed the judges and lawyers to gain insights into the support for our pattern catalog. The interview took place directly after the end of the simulation study with all participants.

## 5 Conclusion

In this paper, we aim to demonstrate how two approaches, design thinking and design pattern, which initially appear to be contrary, can be combined to provide comprehensive support in the development of innovative IT solutions. By first using design thinking for human-centered solution, finding wicked problems, and then implementing the actual solution through design pattern, the support no longer ends before the actual implementation in the system but rather the developers will be further supported.

Design thinking usually ends with the prototypical design of the design solution and therefore does not offer any support in the practical implementation. We see two possible stages in which design pattern can be used in design thinking and thus the potential of both approaches can be elaborated. First, during the analysis of the problem and the identification of possible solutions. Design patterns show the participants the possible solution space and thus support their creativity. Second, the outcome of design thinking is the prototypical implementation and its evaluation of a design solution. Design pattern provide developers beyond that with support in the practical implementation of the design solution. By using the design pattern in the final design thinking stages, we see several advantages:

- Creativity is enhanced by presenting new ideas in the design pattern.
- It is possible to think about which design solutions are realistic at an early stage.
- The result of design thinking is not *only* the design solution but also the corresponding design pattern used to implement the design solution.
- Law regulation rules can be considered early in the development process, thus GDPR compliance can be achieved.

The results of the evaluation also confirm this and show that the combination of both approaches leads to IT artifacts that produce good results both in user evaluation and in court. Further work should analyze the combination of design thinking and design pattern in more detail to expand the deployment scenarios of both approaches. Our contribution is intended to lay a foundation for exciting discussion on the extension and application of design thinking beyond the previously known context.

## References

1. Knote, R., Janson, A., Söllner, M., Leimeister, J.M.: Value Co-Creation in Smart Services: A Functional Affordances Perspective on Smart Personal Assistants. Journal of the Association for Information Systems (JAIS) (2021)

2. Hobert, S., Wolff, R.M. von: Say Hello to Your New Automated Tutor–A Structured Literature Review on Pedagogical Conversational Agents. ICIS (2019)

3. Winkler, R., Hobert, S., Salovaara, A., Söllner, M., Leimeister, J.M.: Sara, the Lecturer: Improving Learning in Online Education with a Scaffolding-Based Conversational Agent. In: Bernhaupt, R., Mueller, F.'F.', Verweij, D., Andres, J., McGrenere, J., Cockburn, A., Avellino, I., Goguey, A., Bjørn, P., Zhao, S. et al. (eds.) Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, pp. 1–14. ACM, New York, NY, USA (04212020)

4. Winkler, R. & Söllner, M.: Unleashing the Potential of Chatbots in Education: A State-Of-The-Art Analysis.

5. Hehn, J., Uebernickel, F.: Towards an understanding of the Role of Design Thinking for Requirements Elicitation-Findings from a Multiple-Case Study. (2018)

6. Alexander, C.: A pattern language: towns, buildings, construction. Oxford University Press (1977)

7. Borchers, J.: Teaching HCI design patterns: Experience from two university courses. CHI (2002)

8. Koukouletsos, K., Khazaei, B., Dearden, A., Ozcan, M.: Teaching Usability Principles with Patterns and Guidelines, 159–174 (2009)

9. Compagna, L., Khoury, P.E., Massacci, F., Thomas, R., Zannone, N.: How to capture, model, and verify the knowledge of legal, security, and privacy experts. International conference on Artificial intelligence and law, 149–153 (2007)

10. Gamma, E., Helm, Richard, Johnson, Ralph, Vlissides, J.: Design Patterns: Elements of Reusable Object Oriented Software. AddisonWesley Professional (1994)

11. Wania, C.: Exploring Design Patterns as Evaluation Tools in Human Computer Interaction Education. MWAIS (2019)

12. vom Brocke, J., Winter, R., Hevner, A., Maedche, A.: Accumulation and evolution of design knowledge in design science research – A journey through time and space. Journal of the Association for Information Systems (JAIS) 23, 9–49 (2020)

13. Nonaka, I., Takeuchi, H.: The Knowledge-Creating Company. How Japanese Companies Create the Dynamics of Innovation. Oxford University Press (1995)

14. Dickhaut, E., Li, M.M., Janson, A., Leimeister, J.M.: Developing Lawful Technologies – A Revelatory Case Study on Design Patterns. HICSS 54 (2021)

15. Dickhaut, E., Janson, A., Leimeister, J.M.: The Hidden Value of Patterns – Using Design Patterns to Whitebox Technology Development in Legal Assessments. 16th International Conference on Wirtschaftsinformatik (WI 2021) (2021)

16. Dickhaut, E., Miedzianowski, N., Jandt, S., Janson, A., Knote, R., Leimeister, J.M., Roßnagel, A., Söllner, M., Thies, L.F.: Handlungsbroschüre. Anforderungs- und Entwurfsmuster zur rechtsverträglichen und qualitätszentrierten Gestaltung kontextsensitiver Applikationen (AnEkA). Handlungsempfehlungen zur Gestaltung von Entwurfsmustern. Universität Kassel (2020)

17. Dickhaut, E., Janson, A., Leimeister, J.M.: Codifying Interdisciplinary Design Knowledge through Patterns – The Case of Smart Personal Assistants. DESRIST (2020)

18. Roßnagel, A., Schuldt, M.: The Simulation Study as a Method of Evaluating Socially Acceptable Technology Design, 108–116 (2013)

19. Thies, L.F., Dickhaut, E., Janson, A., Roßnagel, A., Leimeister, J.M., Söllner, M.: Die Simulationsstudie als Evaluationsmethode. Interdisziplinäre Evaluation eines smarten persönlichen Assistenten. Datenschutz und Datensicherheit (DuD) (2020)