

Please quote as: Dickhaut, E.; Janson, A.; Söllner, M.; Leimeister, J. M. (2023).
Lawfulness by design – development and evaluation of lawful design patterns to
consider legal requirements. *European Journal of Information Systems (EJIS)*.

Lawfulness by design – development and evaluation of lawful design patterns to consider legal requirements

Ernestine Dickhaut, Andreas Janson, Matthias Söllner & Jan Marco Leimeister

To cite this article: Ernestine Dickhaut, Andreas Janson, Matthias Söllner & Jan Marco Leimeister (2023): Lawfulness by design – development and evaluation of lawful design patterns to consider legal requirements, European Journal of Information Systems, DOI: [10.1080/0960085X.2023.2174050](https://doi.org/10.1080/0960085X.2023.2174050)

To link to this article: <https://doi.org/10.1080/0960085X.2023.2174050>



© 2023 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



[View supplementary material](#)



Published online: 01 Mar 2023.



[Submit your article to this journal](#)



[View related articles](#)



[View Crossmark data](#)

Lawfulness by design – development and evaluation of lawful design patterns to consider legal requirements

Ernestine Dickhaut ^a, Andreas Janson ^b, Matthias Söllner ^c and Jan Marco Leimeister ^{d,a,b}

^aInformation Systems, University of Kassel, Research Center for IS Design (ITeG), Kassel, Germany; ^bInstitute of Information Management, University of St. Gallen, St Gallen, Switzerland; ^cInformation Systems and Systems Engineering, University of Kassel, Research Center for IS Design (ITeG), Kassel, Germany

ABSTRACT

New political objectives, emerging regulatory regimes for the digital sphere, and higher penalties for violations have intensified the pressure to develop lawful IT artefacts. As the adaptation of existing IT artefacts to new regulations can be expensive and arduous, a more attractive approach would be to design IT artefacts lawfully from the beginning. A major challenge is that the law is generally technology-neutral, and lawful design requires legal expertise throughout the development, which is costly and time consuming due to communication challenges between legal experts and developers. One possible approach to proactively consider IT regulations in the systems development is design patterns that convey legal design knowledge and support developers in determining the appropriate design options. Consequently, we develop a framework for lawful design patterns and demonstrate their feasibility and advantages using the example of developing AI-based assistants and the regulation of the General Data Protection Regulation (GDPR). Using the design pattern framework, we develop design patterns for lawful AI-based assistants and evaluate them using (a) an experimental approach to show the usefulness of the patterns for developers and (b) rely on a legal simulation study to holistically evaluate how design patterns contribute to lawful IT.

ARTICLE HISTORY

Received 29 October 2021
Accepted 20 January 2023

KEYWORDS

Design patterns; legal requirements; codifying design knowledge; law simulation study; GDPR; AI-based assistants; design science research

1. Introduction

The protection of one's personal data and related legal aspects are becoming increasingly salient in light of recent privacy scandals (Baruh et al., 2017). Especially the European area is well-known for its strict regulations and standardisations (Büthe & Mattli, 2013; Martin & Matt, 2018), i.e., through the General Data Protection Regulation (GDPR), which came into effect in May 2018 (European Union, 2018). The GDPR is a data protection legislation that regulates the processing, storing, and managing of data from people who are protected by the GDPR. The GDPR has a global impact on companies that interact with the European market or provide services to EU customers, as it is not geographically limited to the EU but protects EU citizens worldwide (Li et al., 2019; Peukert et al., 2022).

Lax consideration of legal requirements often leads to violations of the law, resulting in expensive adjustments with the aim of meeting Europe's strict requirements (Ayala-Rivera & Pasquale, 2018). According to PricewaterhouseCoopers (PwC, 2017), more than three quarters of American companies are expected to spend between \$1 million and \$10 million to comply with the new GDPR, and related fines have recently increased. To date, for

example, 198 fines totalling €163,815,398 have been imposed for data processing without a sufficient legal basis (CMS Legal, 2021; Ford et al., 2021). To avoid GDPR-related penalties, providers of video conferencing systems, such as Zoom (Security Week, 2020), and AI-based assistants, such as Amazon's Alexa (Furey & Blue, 2018; Lau et al., 2018), have had to make ex-post adjustments due to the strict European requirements. For example, Zoom majorly violated the GDPR in the huge European market, which ultimately led to negative publicity, and it was obligated to respond with extensive ad hoc changes to its system functionalities and large revisions of its privacy statements (Security Week, 2020; Vanberg, 2021). To avoid such ad hoc changes in the later stages of systems development or related fines for violations, developers must consider legal requirements as early as possible to ensure lawful solutions (Acquisti et al., 2022). In this context, lawfulness means that the legality of a system meets the legal requirements to be approved for the market as a key principle of the GDPR (Aljerisy et al., 2021). However, regulations are often formulated vaguely, which makes it difficult for practitioners to extract and operationalise legal requirements from the

CONTACT Ernestine Dickhaut  ernestine.dickhaut@uni-kassel.de; Andreas Janson  andreas.janson@unisg.ch  Information Systems University of Kassel, Research Center for IS Design (ITeG), Kassel, Germany

 Supplemental data for this article can be accessed online at <https://doi.org/10.1080/0960085X.2023.2174050>.

© 2023 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

GDPR (Ayala-Rivera & Pasquale, 2018). In addition, developers often struggle to find adequate design solutions, since the law is generally technology-neutral and must be considered for each individual case (Hadar et al., 2018; Hildebrandt & Tielemans, 2013). Legal requirements thus increase pressure on developers of systems who often lack the necessary domain expertise to implement regulation rules or to develop lawful IT artefacts (Smith & McKeen, 2006; van der Sype & Maalej, 2014). As a result, uncertainties regarding practical implementation often lead to situations in which legal requirements are implemented with minimal attention so that the system just about satisfies them (Aljeraisy et al., 2021). Thus, responding *ex ante* to regulation is becoming increasingly important, for example, the European Commission recently released the Digital Markets Act (DMA) and the Digital Services Act (DSA), which again introduce new regulations that need to be considered (Laux et al., 2021; Petit, 2021).

Design patterns originated in architecture as a means of solving recurring problems (Alexander, 1979) and have long been established in systems development (Gamma et al., 1994; Wania, 2019). Design patterns are proven solutions for recurring problems that codify complex domain knowledge in an accessible and applicable way for non-domain experts. Thus, design patterns offer a promising solution to recurring legal requirement problems and may eliminate uncertainties affecting the implementation of legal requirements. The design pattern development process can be either inductive or deductive (Petter et al., 2010). To support developers in developing lawful IT artefacts, we present a design-pattern-based solution that integrates legal requirements early on in the development of IT artefacts by making regulation rules accessible to developers and providing proven solutions for implementing regulation rules. Using a deductive approach, design patterns could present proven design solutions by concretising European law to solve legal design issues (Hoffmann et al., 2015). Owing to the specification of law through fundamental rights and legislation, legal requirements stabilise over time and occur repeatedly in many development contexts in a similar form (Hoffmann et al., 2015). Thus, a framework to guide the codification of legal design knowledge through design patterns could support researchers and practitioners. Consequently, in this paper, we address the following research questions (RQ):

RQ 1: How should legal design knowledge be codified in a design pattern framework to make resulting design patterns useful for the development of lawful systems?

RQ 2: To what degree does the application of the design pattern framework and the relevant design patterns lead to the development of lawful systems?

To answer our RQs, we follow a design science research (DSR) approach to develop a design pattern framework that supports researchers and practitioners in codifying legal design knowledge through design patterns (RQ1). We describe the pattern framework development in which we consider the cognitive processes of the primary pattern users (i.e., the system developers) using cognitive load theory and theoretical insights from pattern development and legal research. We apply the design pattern framework in the case of developing lawful AI-based assistants and develop design patterns using the framework. Afterwards, we evaluate the design patterns with a multi-method approach including developers and legal experts (RQ2). First, we evaluate the design patterns in the field with developers to draw conclusions as to whether they are suitable to support developers in the process of designing lawful systems. Second, we assess the design patterns with judges and lawyers in simulated court cases in terms of their lawfulness. Thus, we include legal experts in our evaluation process, since they have the necessary knowledge to decide about how our developed design patterns improve the system's lawfulness. We contribute to the body of knowledge with a nascent design theory of legal design by extending the known benefit of design patterns to the domain of IT regulation and showing how to effectively improve lawful system design. In addition, we contribute to resolving the lack of understandable and applicable legal rules.

2. Conceptual background

2.1. Regulation of IT artefacts

IT regulation has a sudden and considerable effect on organisations and the public (Smith & McKeen, 2006; Väyrynen & Lanamäki, 2020), especially when considering privacy aspects (Lowry et al., 2017). Stricter regulation rules arising from increased interest in personal data protection govern the processing, storage, and management of data. For example, Europe's GDPR has had a global impact on companies that interact with the European market but also beyond that. These regulatory spillovers are described as the Brussels effect, i.e., providers of services even outside of the EU and not catering to EU citizens try to comply with the GDPR (Peukert et al., 2022). The same holds true for other data protection regulations that are currently proliferating to protect users (Aljeraisy et al., 2021; Peukert et al., 2022). Thus, sufficient attention must be paid to legal requirements early on

and systematically during systems development to avoid lawsuits, negative publicity, and high costs associated with revising systems accordingly. Consequently, systematic approaches to designing novel systems that comply with standards, such as privacy by design, are becoming increasingly important to regulate systems development early on (Hadar et al., 2018).

Ideally, legislators and regulators would publish regulations and rules in an unambiguous, easy-to-interpret, human-readable, and machine-readable format (Butler, 2017). In practice, however, legal rules are often equivocal and must first be transformed and made explicit (Weick, 2010). The involvement of legal experts in the early stages of the development of novel IT artefacts can help to prevent the violation of laws (Roßnagel & Schuldt, 2013). However, involving legal expertise is often problematic, since experts are costly and communication between legal experts and laypersons is often challenging, since it mostly relies on texts using legal jargon (Becker et al., 2014; Knackstedt et al., 2014). Organisations face legal requirements with complex challenges that go far beyond the implementation. Organisations rarely focus on effective privacy strategies because personal data is a key element for many companies' business models (Spiekermann, 2012). Ultimately, the responsibility for implementing all regulations and guidelines tends to fall, at least partially, on the shoulders of the developer (Smith & McKeen, 2006). Vaujany et al. (2018) recognise the key challenge of making rules consistently meaningful, regardless of whether the rule is visibly or invisibly entangled in IT. Oftentimes, legal rules that are implemented using IT become invisible to the user.

IS research on IT regulation primarily examines the regulation's impact on innovation in the IT market (see, e.g., Guerra & Koh, 2019; Martin & Matt, 2018) and the consequences for development and the user (see, e.g., Huth et al., 2020; Väyrynen & Lanamäki, 2020), or integrates legal requirements in development through making models (Becker et al., 2014; Siena et al., 2009). Vaujany et al. (2018) distinguish three streams of IS research that focus on IT-based regulation: (1) research on mechanisms of integrating regulation rules into IT artefacts and the processes'

properties, (2) practical sensemaking and the adherence of rules, and (3) the alignment of IT artefacts and practices complying with the rules.

According to the overall research background (Table 1), we contribute to resolving the lack of understandable and applicable legal rules in systems development. We focus on one specific class of artefacts, namely AI-based assistants, to demonstrate the integration of design patterns in early development. This allows us to investigate how legal rules can be made applicable for developers.

AI-based assistants have recently been subject to strict regulation owing to their data processing, storage, and management potential. These systems, such as Amazon's Alexa, are proliferating in daily life as ubiquitous digital service platforms. In this context, developers often lack the necessary legal expertise to successfully implement the legal requirements. Consequently, legal requirements are often only minimally addressed (Hoffmann et al., 2015), and the focus of development lies on other types of requirements such as user experience. To satisfy all requirements, developers must consider how AI-based assistants can be lawfully designed while still meeting the user's other requirements such as service quality expectations. In the next subsection of our paper, we introduce the concept of design pattern as a means for codifying legal design knowledge.

2.2. Design patterns to codify legal design knowledge

Design patterns were originally introduced by Alexander (1977) in architecture. In systems development, design patterns have become an established approach used to codify design knowledge. Design patterns document known and proven solutions to recurring problems (Gamma et al., 1994) by including templates to describe information in tabular form and representing established instruments to make complex knowledge accessible and applicable (Alexander, 1977). In systems development, patterns were first established by the "Gang of Four" (Gamma et al., 1994), and their use of patterns has become established in various disciplines. In human – computer

Table 1. Research Background.

Study	Key Results
Smith and McKeen (2006)	Explores how new regulation rules change IT work and the resulting regulatory issues
Weick (2010)	Regulation rules are equivocal, must first be transformed, and made explicitly
Almeida et al. (2020)	Developed a framework to provide a lens for what about, when, and how the use of AI should be regulated
Vaujany et al. (2018)	IT-based regulation systems consist of rules, practices, and IT artefacts and their relationships as a trifecta
Blind et al. (2017)	In situation of high market uncertainty, regulation leads to lower innovation efficiency
Butler (2017)	Identifies challenges of RegTech: regulation-interpretation problem, and absence of a common language
↓	
This study	Makes abstract regulation rules understandable and applicable for developers through design patterns
↑	
Becker et al. (2014)	Meta-design for integrating regulations into the design of IS

interaction (HCI), several studies have demonstrated the successful use of patterns in teaching design principles and design concepts (Borchers, 2002, April; Compagna et al., 2009; Koukouletsos et al., 2009). Design patterns have also received attention in IS research to codify design knowledge (Tuunanen & Holmström, 2021).

Design patterns can further enable a broad understanding of peripheral disciplines, i.e., approaches that map legal knowledge into patterns (Hoffmann et al., 2015; van der Sype & Maalej, 2014; Yskout et al., 2015). Challenges to the implementation of legal requirements include frequently recurring problems. In the European legal system, the foundation of law relates to fundamental rights and legislation that make the law relatively stable over time. Thus, IT regulation can occur repeatedly in many development contexts in similar forms. Hence, design patterns represent a promising approach for the integration of regulation rules in the design of novel artefacts and their uncertainties in IT implementation.

3. Design journey

3.1. Methodology

We follow Peffers et al. (2007) DSR process to develop our design pattern framework as well as to demonstrate and evaluate the framework through application in one specific case to develop design patterns for lawful AI-based assistants (see Figure 1). Specifically, we follow a problem-centred approach that relies on developers' support to proactively consider IT regulation early in the systems development process.

We conducted a focus group workshop ($n = 8$) to

justify and complement the requirements derived from literature and to include practical insights in our design pattern framework (see Table 2). The participants in the workshop were either developers or legal experts (please find the participant details in "Appendix A"). Based on the requirements, we derived pattern elements that we used to develop a design pattern framework. We evaluated the design patterns in two evaluation settings, resulting in two revisions of the design patterns. The iterative approach allows us to evaluate the design pattern framework and the design patterns in depth and to improve them with theoretical and practical feedback.

3.2. Requirement derivation for the design pattern framework

We draw on extant research findings to codify legal design knowledge. In particular, we use research regarding the visualisation and codification of design knowledge (Chandra Kruse & Nickerson, 2018; Schoormann et al., 2021; Vom Brocke et al., 2020) but in particular also literature on the specifics of legal knowledge (Becker et al., 2014; Knackstedt et al., 2014; Vaujany et al., 2018), visual inquiry tools (Osterwalder & Pigneur, 2013), and cognitive load theory (CLT) as our kernel theory to develop our design pattern framework.

CLT provides a theoretical framework for how individuals process information during learning and problem-solving processes while guiding the structuring of information for better learning results (Sweller, 1988). We argue that the use of design patterns in interdisciplinary IS development is also a problem-

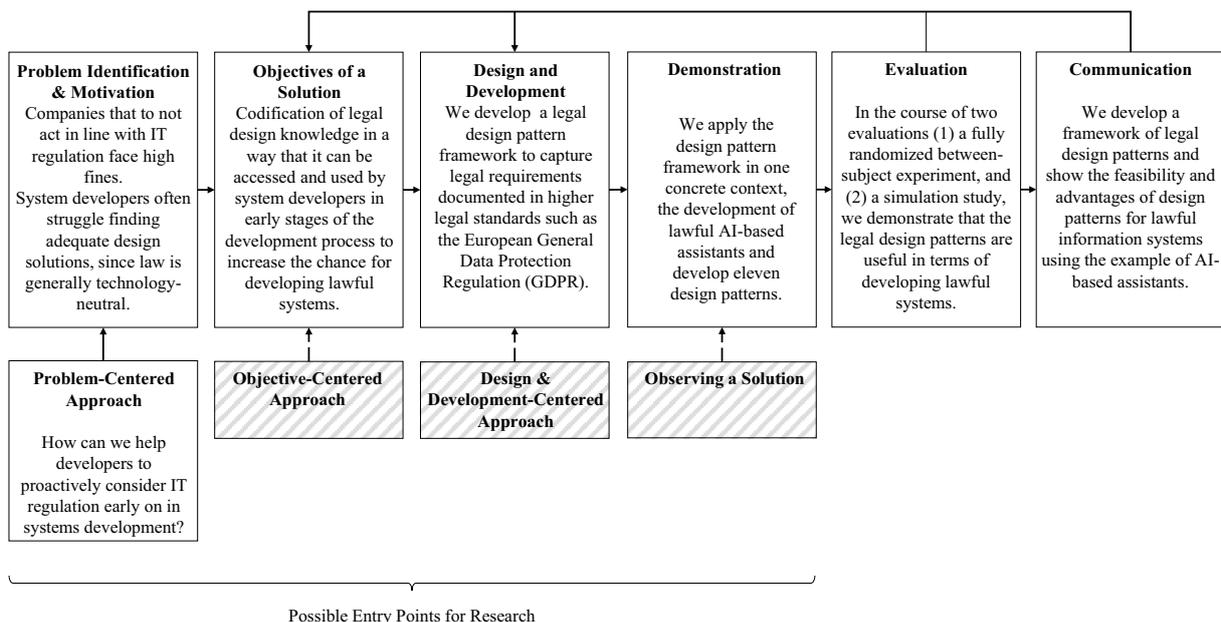


Figure 1. Design science research approach adapted from Peffers et al. (2007).

Table 2. Lawful design pattern requirements.

	Requirements	Legal Relevance	Source
R1	Related information should rather be presented together to avoid split-attention effect and to reduce extraneous load resulting from complex legal knowledge.	Connecting legal and technical requirements to avoid split-attention effects.	Aljeraisy et al. (2021); Ayres and Sweller (2005)
R2	Information needs to be appropriately structured (e.g., through a clear subdivision of legal terms) so that different subjects are clearly perceptually distinguished and allow rapid retrieval of legal information.	Breaking down legal texts to make them accessible	Figl (2017); Kirschner et al. (2011); Malinova and Mendling (2013); Osterwalder and Pigneur (2013)
R3	A design pattern should communicate why it is useful to use it to clarify the importance to implement the legal requirement(s) and the related added value.	Communicating the added value to increase the perceived importance	Garud (1997); Ayala-Rivera and Pasquale (2018)
R4	Legal knowledge should be formulated compact and context-specifically adapted to its use case scenario to overcome technology neutrality of law.	Formulating compact and context-specific legal knowledge to overcome technology neutrality of law	Legner et al. (2020); Ayala-Rivera and Pasquale (2018)
R5	Include requirements, the relationship between constructs and domain objects, an explanation, and a justification of the requirements.	Providing information behind the legal requirements	Heinrich and Schwabe (2014); Hadar et al. (2018)
R6	A design pattern should provide additional domain-specific information and knowledge to support the application.	Supporting the transfer into practical implementation	Chandra Kruse and Seidel (2017)
R7	A design pattern should provide implementation guidance but should not restrict the pattern users in their creativity.	Legal requirements may be prioritised higher due to their relevance and the risk of penalties, possibly affecting the developer's creativity	Ahrens and Sankar (1993)
R8	A design pattern should provide an overall understanding of the artefact to be designed to successfully integrate legal requirements.	Considering the overall artefact design during the implementation of legal requirements	van Aken (2005); Walls et al. (1992); Ayala-Rivera and Pasquale (2018)
R9	The design pattern should enable knowledge transfer, document it, and provide good repositories for legal knowledge dissemination.	Providing a legal design knowledge repository	Chandra Kruse and Seidel (2017); Huth et al. (2020)
R10	The codified design knowledge should be abstracted to make it usable for a class of artefacts.	Balancing legal requirement abstractions	Nonaka and Toyama (2003); Baxter et al. (2007)
R11	Besides a solution the design pattern should support the understanding of the legal problem context and the underlying legal requirements.	Understanding the underlying legal mechanism	Rothe et al. (2020); Vom Brocke et al. (2020); Compagna et al. (2009); focus group workshop

solving process wherein developers acquire design knowledge and apply it to complex problems (i.e., when disputes arise from conflicting requirements).

In CLT, it is typically assumed that there is a fixed number of cognitive resources in the working memory that an individual is able to invest in a task, e.g., systems development tasks. When considering the main tenets of CLT (Kalyuga, 2011), we mainly distinguish between two types of loads¹: intrinsic and extraneous load. Intrinsic load represents a task's inherent difficulty (Kalyuga et al., 2009). By contrast, any other load that does not promote learning and problem-solving is considered extraneous (Kirschner et al., 2011), determined by the manner and

complexity in which material relevant to the task is presented. Both types of loads are additive in nature and determine the total load imposed by the task and material that needs to be processed when solving a task. However, if the cognitive load exceeds the available resources of the working memory, the cognitive system will fail to process relevant information. When transferring these mechanisms to the task of systems development, CLT seeks in our case to manage cognitive load in a way that there is a shift on how developers invest their overall cognitive resources. Following this thought, it is not the goal to reduce the overall cognitive load but rather to acknowledge the cognitive processes and enable developers to deal

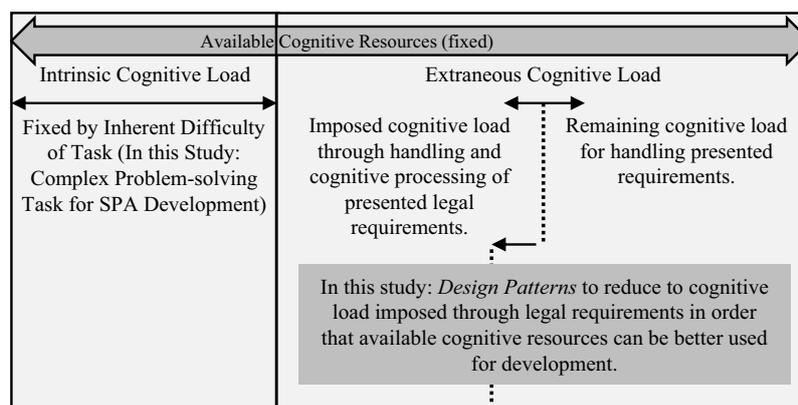


Figure 2. Cognitive load theory application.

more easily with legal aspects and invest more cognitive resources in the systems development (see Figure 2).

Following these thoughts, the integration of different information during systems development processes increases extraneous cognitive load (Ayres & Sweller, 2005). Therefore, extraneous load can be controlled through presenting related information together. Legal guidelines are oftentimes disconnected from technical requirements, which is the reason why developers oftentimes face the problem of adapting them to implementations that properly meet privacy requirements (Aljeraisy et al., 2021). Through the spatial and content combination of technical and legal knowledge, the split-attention effect is prevented (R1).

A long continuous text of codified information – as seen in legal documents or laws oftentimes – and the technology neutrality of the law complicate its application. As such, there is no overview, and the possibility of extracting the required information as quickly as possible is missing. There is various research on investigating a clearly arranged presentation form of codified information, such as the Business Model Canvas as an organising device for designing business models (Osterwalder & Pigneur, 2013). As an individual's cognitive resources are limited, it is essential to manage the cognitive load related to the pattern presentation when taking design advice offered by design patterns into account (Kirschner et al., 2011). Visual inquiry tools (Avdiji et al., 2020) demonstrate how a clear and well-arranged layout makes it possible to present the essence of design information in a rather small space, which we use for the development of helpful design patterns to achieve lawful artefacts. Designing lawful artefacts is a complex procedure that includes balancing different requirements and interpreting laws. Thus, the separate information of a design pattern should be adjusted by a clear subdivision to achieve perceptual discriminability through structuring information appropriately (Figl, 2017; Malinova & Mendling, 2013). Such a clear subdivision prevents confusing design knowledge representations as well as a time-consuming search of the necessary details in legal texts (R2).

During systems development, legal requirements are often only addressed to a minimum extent in order to be compliant with the minimal requirements of law (Hoffmann et al., 2015). However, the protection of one's data is also important, especially when taking new legal regulations and user fears into account (Barati et al., 2019). In this context, system developers often lack the necessary domain expertise to be able to implement legal requirements for developing a lawful IT artefact. A design pattern requires the designer to understand the pattern's intention and purpose,

function, and mechanism (Gamma, 1995). Thus, the design patterns' content should communicate the added value resulting from the pattern use and the consideration of legal requirements (Ayala-Rivera & Pasquale, 2018) (R3).

Any obfuscation or lack of clarity will diminish the pattern's comprehensibility and benefit (Taylor, 2001). Clear presentation increases patterns' quality and usefulness for users (Doering & Veletsianos, 2007). Legner et al. (2020), for instance, highlight how important it is for codified design knowledge presentation to be as compact and clear as possible to clarify the meaning of the solution. Law and legal requirements are often vaguely formulated and leave room for interpretation. This is further reinforced by the technology neutrality of the law. Therefore, legal knowledge should be formulated as specifically to each context as possible and applied to the use case (Ayala-Rivera & Pasquale, 2018) (R4).

Designing lawful IS is an interdisciplinary task including several, sometimes conflicting requirements. To codify complex information such as legal knowledge, the design pattern approach should include requirements, the relationship between constructs and domain objects, an explanation, and a justification (Heinrich & Schwabe, 2014). An integrated perspective on regulating IT that considers the relationships between rules, IT artefacts, and practices is still lacking (Vaujany et al., 2018) (R5). Developers' experiences, instincts, and intentions, which are often generated by experiences, are valuable information for designing artefacts. Instinct is developed through numerous applications of solutions but is difficult to explicate and codify because it defies brief description and must be considered in context (McLeod & MacDonell, 2011). These years of experience can be passed on through additional domain-specific information to provide insights into the special features of the artefact class (R6). However, a pattern should provide implementation guidance for solving recurring problems but should not restrict pattern users' creativity (Ahrens & Sankar, 1993). Design patterns' content should help the user to expand their knowledge and to learn to solve the following problems (Garud, 1997) (R7).

Knowledge about the artefact (to be designed) is a crucial feature of design knowledge. To develop an artefact, knowledge about the characteristics and properties of the artefact and its materials is required (van Aken, 2005). Walls et al. (1992) describe object knowledge using "meta-design", which is intended not for a specific artefact but for a class of artefacts. If developers are not even familiar with the technical possibilities, they (probably) cannot weigh up or integrate the legal requirements. Mapping legal

requirements onto software functionality is nontrivial and requires an overall understanding of the artefact to be designed (Ayala-Rivera & Pasquale, 2018) (R8).

Chandra Kruse and Nickerson (2018) assume that design knowledge is codified for reuse, and although reuse is sometimes associated with repetitiveness, it has been observed in contexts that strive towards innovation and customisation. This could be especially useful for organising legal knowledge that is often based on stable basic rights. Thus, the core content of the design pattern should enable knowledge transfer, document it, and provide good repositories for legal knowledge dissemination (R9).

The applicability and reusability of design knowledge depends on the degree of abstraction. While abstract knowledge can be applied to many different scenarios, there are only limited concrete solution instructions (Elshan et al., 2022), while concrete knowledge is only useful in a small circle of applications. Therefore, the design patterns' level of abstraction must be well considered (Baxter et al., 2007; Nonaka & Toyama, 2003). Ideally, the codified design knowledge should be abstracted to make it usable for a class of artefacts (R10).

In the end, the design of IS is always a problem-solving process that is only successful if the problem is truly understood (Vom Brocke et al., 2020). To help

the user understand the problem, the problem context should be codified along with the solution. In practice, IT regulation rules are often unclear and must first be made explicit (Weick, 2010). Ultimately, the implementation of all regulations and guidelines tends to fall – at least partially – to the developer (Smith & McKeen, 2006). Once the IT leads to obvious violations – for example, through non-compliance with the GDPR – the regulatory violations can be taken to court. Therefore, the gap between rule creation, its implementation in IT, and rule adherence has widened, making it difficult to determine whether a rule is being followed and what the consequences are if not (Vaujany et al., 2018). A notable gap persists regarding new demands for and ways of regulating IT. Thus, it is necessary to provide information about the legal problem context and the underlying legal requirements in order to support the developer in understanding and applying the legal requirements (R11).

3.3. Design and development of the design pattern framework

As indicated in the section above, we identified eleven requirements for developing our design pattern framework by extracting lawful design-relevant

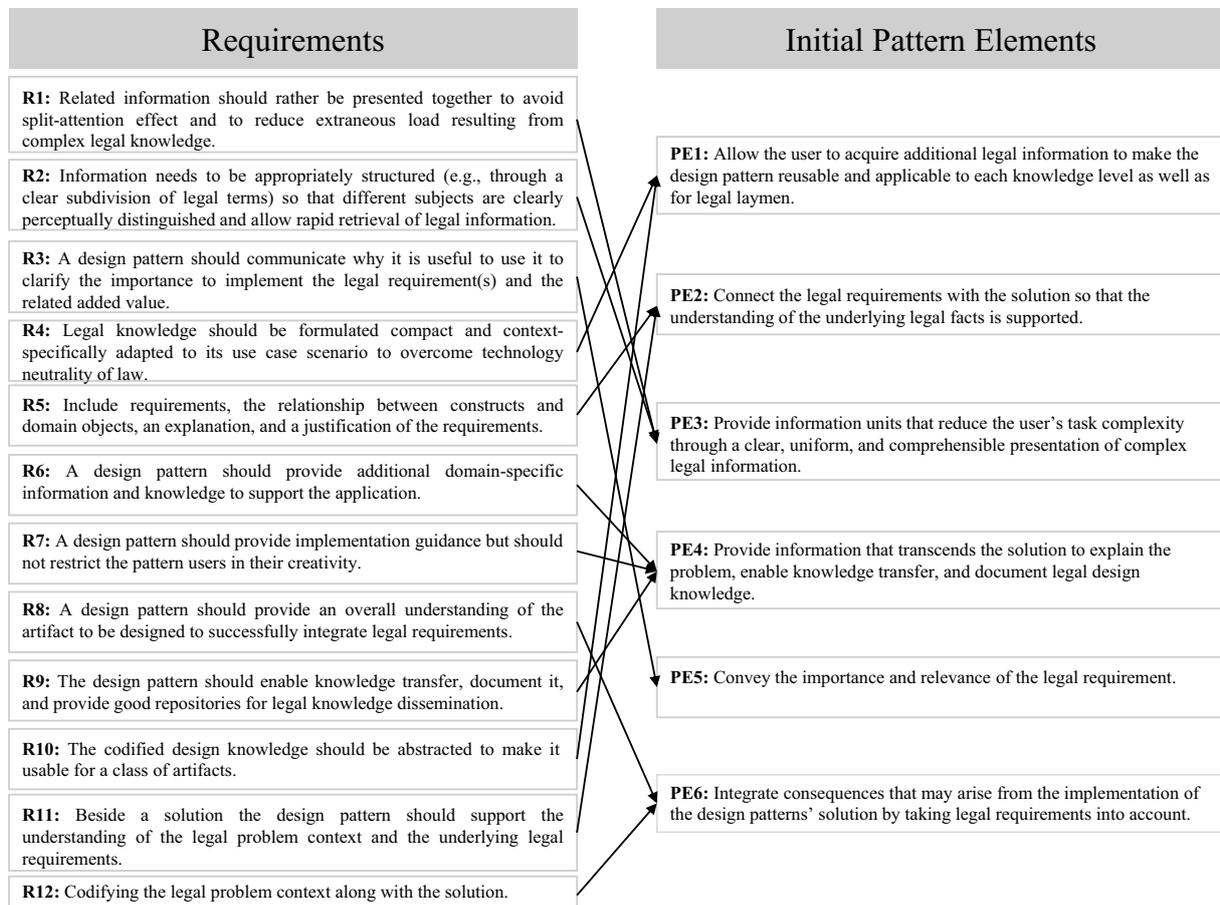


Figure 3. Mapping of the requirements to the initial pattern elements.

information from previous work. Moreover, we based our requirements on literature concerned with scaffolding problem-solving, which has also proven useful in managing cognitive load (Sweller, 1988). We use the requirements to develop a pattern-based approach that integrates legal requirements early in the artefact's development. In the following, we describe how we derive the pattern elements and how they manifest in our initial design pattern framework (see Figure 3).

Software developers are usually not legal experts. Nevertheless, they have to implement legal requirements. Depending on additional training or experience, developers develop legal knowledge or understanding. To ensure that our design patterns offer the highest possible added value for multiple users, special attention must be paid to the different levels of legal knowledge developers have. The design incorporates both technical knowledge and creativity and occurs as a reflective conversation between designers, their actions, and their situations (Chandra Kruse & Seidel, 2017). Developers often lack the legal knowledge to find satisfactory solutions. Regulations and legal requirements are considered to be necessary for market approval, but the benefits and added value are often not understood. Thus, we suggest our first pattern element:

Pattern Element #1: Allow the user to acquire additional legal information to make the design pattern reusable and applicable to each knowledge level as well as for legal laymen.

The first pattern element indicates the need to provide the user with further information. Legal knowledge can be particularly complex and includes specialist terms, depending on the context. Thus, the pattern user may obtain further information depending on the legal knowledge level and the application context. In our initial design pattern, we implement this element by linking related design patterns and including links to more detailed explanations of the implementation.

Finding legal design solutions can be challenging and requires an extensive understanding of the artefact being developed as well as the understanding behind the legal requirements. To support the user in solving a development problem, a concrete example may help (Chandra et al., 2014). By including further content and information, not only is the comprehensibility of the problem and the possible solution improved but the user's expertise is also increased, allowing them to solve related problems and work on domain knowledge shortcomings (Janson et al., 2020). This results in our second pattern element:

Pattern Element #2: Connect the legal requirements with the solution so that the understanding of the underlying legal facts is supported.

We structured the solution area to provide an overview of the application of the pattern first, and then, if

necessary, further information can be accessed via linkable explanations. To support the user's understanding of the legal background, a description of the relevant legal requirement is included. This description focuses on the artefact's prospective user.

The content is not only of great importance for the pattern's applicability and added value but also for how the design knowledge is presented. Legal knowledge is often couched in technical jargon, making it difficult to use (Geradin et al., 2021). In the last years GDPR has proven to be too intricate and oftentimes hard to apply in real life (Saqr, 2022). In addition, IT development is characterised by stress and a lack of time. Therefore, developers need support that provides suitable solutions in the shortest possible time. A clear presentation method should thus be used to manage the cognitive process (Doering & Veletsianos, 2007) by reducing the split-attention effect through presenting related information as a unit. Thus, we suggest our third pattern element:

Pattern Element #3: Provide information units that reduce the user's task complexity through a clear, uniform, and comprehensible presentation of complex legal information.

Building upon the thoughts of the third pattern element, we follow the presentation of the design knowledge as a one-pager (Hsiao & Lopez, 2016; Osterwalder & Pigneur, 2013). A clear presentation is also utilised by other concepts from systems development such as "cheat sheets" (Wang et al., 2020). The information comprises a mixture of bullet points and sentences, facilitating a clear presentation. Thereby, the pattern content is presented in a spatial combination of related information.

Legal requirements are a constant companion in today's software development. Ideally, developers should gain experience in the implementation of legal requirements over time and be able to transfer these to new contexts. Information that allows the user to understand the problem's context (Chandra et al., 2014) and critical problem-solving aspects (Reiser, 2004) should be provided. In this context, a design pattern should provide information about the artefacts' actions and, according to boundary conditions, enable knowledge transfer (Seidel et al., 2018). The design patterns should provide implementation guidance and must be generally applicable (Gamma et al., 1994). The pattern's content should not be a concrete solution to a problem; rather, it should help the user to expand their knowledge and learn to solve the problems. Hence, we propose:

Pattern Element #4: Provide information that transcends the solution to explain the problem, enable knowledge transfer, and document legal design knowledge.

We integrate additional information about the pattern's solution, requirements, and influences to support the user's understanding of the problem and

enable knowledge transfer (Zahedi & Babar, 2014). Moreover, we use the section “current period of development process” to show at which point in the development process the pattern is useful to guide the pattern user during the selection process.

The implementation of legal requirements is often a necessary burden for developers but one that is implemented rather reluctantly and with little effort in the end. Therefore, the added value through the implementation of the design pattern should be communicated. Thus, we suggest our fifth pattern element:

Pattern Element #5: Convey the importance and relevance of the legal requirement.

Legal requirements usually have an effect on the entire system and its design and operation. Thus, the

application of the design pattern solution can impact other components of the IT artefact. In particular, if a user is inexperienced, the design pattern should critically address possible effects. Therefore, legal knowledge should be seen in context. Hence, we propose:

Pattern Element #6: Integrate consequences that may arise from the implementation of the design patterns’ solution by taking legal requirements into account.

Since solutions and legal requirements often impact the entire artefact in practice, this impact should also be outlined in the design pattern. For this purpose, we integrate the category “consequences” in which both the possible positive and negative effects of the pattern’s implementation are listed.

Table 3. Mapping of pattern elements and initial design pattern.

Requirements	Pattern Element (PE)	Implementation into Initial Design Pattern	Field in Design Pattern
R4 Legal knowledge should be formulated compact and context-specifically adapted to its use case scenario to overcome technology neutrality of law.	PE#1 Allow the user to acquire additional legal information to make the design pattern reusable and applicable to each knowledge level as well as for legal laymen.	Linking legal information to support each knowledge level, further information can be accessed via linkable explanations	Relevant legal requirements; Influences
R10 The codified design knowledge should be abstracted to make it usable for a class of artefacts.			
R5 Include requirements, the relationship between constructs and domain objects, an explanation, and a justification of the requirements.	PE#2 Connect the legal requirements with the solution so that the understanding of the underlying legal facts is supported.	Each design pattern includes the laws behind it, ensuring a link between the design solution and the legal requirements	Relevant legal requirements
R11 Beside a solution the design pattern should support the understanding of the legal problem context and the underlying legal requirements.			
R1 Related information should rather be presented together to avoid split-attention effect and to reduce extraneous load resulting from complex legal knowledge.	PE#3 Provide information units that reduce the user’s task complexity through a clear, uniform, and comprehensible presentation of complex legal information.	A clear presentation allows to quickly find the necessary information	Clear arrangements through separated areas; Spatial combination of related information; Current period of development process
R2 Information needs to be appropriately structured (e.g., through a clear subdivision of legal terms) so that different subjects are clearly perceptually distinguished and allow rapid retrieval of legal information.			
R6 A design pattern should provide additional domain-specific information and knowledge to support the application.	PE#4 Provide information that transcends the solution to explain the problem, enable knowledge transfer, and document legal design knowledge.	Integration of additional information about the pattern’s goal, requirements, and influences to support the user’s understanding of the problem and enable knowledge transfer	Solution, Relevant legal requirements; Current period of development process
R7 A design pattern should provide implementation guidance but should not restrict the pattern users in their creativity.			
R9 The design pattern should enable knowledge transfer, document it, and provide good repositories for legal knowledge dissemination.			
R3 A design pattern should communicate why it is useful to use it to clarify the importance to implement the legal requirement(s) and the related added value.	PE#5 Convey the importance and relevance of the legal requirement.	Goal describes how the optimal target state looks like after considering the design pattern	Goal
R8 A design pattern should provide an overall understanding of the artefact to be designed to successfully integrate legal requirements.	PE#6 Integrate consequences that may arise from the implementation of the design patterns’ solution by taking legal requirements into account.	Integration of category “consequences”, in which both the possible positive and negative effects of the pattern’s implementation are listed.	Consequences

Privacy-friendly user profile 		Goal  The user's profile does not contain any sensitive and intimate data. Also, no inferences to sensitive and intimate data are formed and stored in the user profile during the usage process.	
Influences  <ul style="list-style-type: none">• Empathy• Informational self-determination	Solution  <ul style="list-style-type: none">• Specification of profiling data via privacy policy• ► Concrete processing purpose• Generally, no storage of private and sensitive data for profiling (see Art. 9 GDPR)• According to the orientation purpose only data that is explicitly required may be stored in the user profile• Transparency in storage, prohibition, deletion• ► Authorization management• Possibility of ► digital self-disclosure• Regular review of processing purpose, if necessary• ► Deletion routines	Consequences  Law <ul style="list-style-type: none">• Right to “be forgotten”• No reliable classification of sensitive data leads to privacy-unfriendly behavior despite privacy-friendly setting	Current Period of Development Process  <input type="checkbox"/> Interaction Patterns <input type="checkbox"/> Architecture Patterns <input type="checkbox"/> Learning Patterns <input checked="" type="checkbox"/> Data Processing Patterns
		Service Quality  <ul style="list-style-type: none">• No transparency for the user when the system gives inappropriate answers• High implementation effort• Degree of individualization of the system controllable	
Relevant Legal Requirements <ul style="list-style-type: none">• No processing of intimate data• No complete user profile• Setting options for users• Non-chain-ability• Processing of sensitive data only with consent• No discriminatory decisions 			

Figure 4. Initial design pattern.

Thus, we developed six pattern elements as key parts of the lawful design pattern framework. Table 3 provides an overview of the matching between the requirements, pattern elements, and the initial design pattern framework. We use the pattern elements to develop our initial design pattern framework (see Figure 4 for an exemplary design pattern), which we will iteratively evaluate and refine until we have our final design pattern framework. The initial design pattern consists of a unique, meaningful pattern name to describe the key design pattern content. We include a pattern goal description and the “current period of development process” box to guide the pattern user during the pattern selection process. A connection to relevant legal requirements and influences provides necessary domain knowledge to go more into legal details. The patterns’ core is the solution part. Since development oftentimes needs experience and a gut feeling, we integrate consequences to provide an overview of possible effects on other system parts or the whole system. In the following, we

describe our demonstration and evaluation procedure to develop our design pattern framework.

3.4. Demonstration of the design pattern framework

We apply the design pattern framework in one specific case by addressing the challenges in developing IT artefacts, focusing on lawful AI-based assistants as a case to demonstrate the integration of design patterns in development. While lawful AI-based assistants have been on the market for many years, and during this time there have been many legal violations that have led to costly system revisions, it was only in 2021 that the European Data Protection Board (2021) provided official guidelines to support developers concerning the design of these systems. Thus, we apply the design pattern framework and develop design patterns to support the design of lawful AI-based assistants.

AI-based assistants provide daily support in many ways, on smartphones, in cars, in service encounters, in

smart home environments, or as support for elderly or disabled people (Knote et al., 2021). These assistants are an important new IT artefact class and represent smart technical objects that combine contemporary IT artefacts – such as natural language processing, machine learning, and context-sensitive autonomous behaviour – and are often used for smart service provision (Beverungen et al., 2019; Medina-Borja, 2015). An AI-based assistant is defined by mainly using voice and vision as user input and contextual information to provide assistance by answering questions in natural language, making user recommendations, and performing actions (Hauswald et al., 2015). Hence, AI-based assistants offer entirely new ways of engaging users through innovative interaction possibilities with service providers. The pervasiveness of invasive IT artefacts embedded in these systems, as well as their autonomous nature, raises questions of accountability and data security (Cowan et al., 2017), for example, the growing scepticism and concern that these systems “listen” without being activated by a wake word (Lau et al., 2018). The increasing intelligence of AI-based assistants raises further questions on how this new class of systems should be developed in a lawful way. It is not only voice activation that is critical but also the inferences that may be drawn from system interactions, for example, the user’s emotional state might easily be captured without their knowledge, making them vulnerable.

For the design pattern development, we conduct two workshops with developers who have experience in designing AI-based assistants, legal experts and IS researchers. The workshop participants develop the design pattern using literature on AI-based assistants, legal literature and regulation rules, and their own experience (Dickhaut et al. 2020). To demonstrate the design pattern development, one of the developed design patterns, the pattern “Processing Emotional Data”, will be described in more detail in the following. Emotional data are particularly worthy of protection from a legal point of view, as they allow psychological conclusions to be drawn, for example, about the person and can have an impact for other purposes, such as in professional life. AI-based assistants can capture and store such data through voice, commands, and other emotions. This is why processing emotional data is important for lawful AI-based assistants. From the GDPR perspective Art. 5 Para. 1 lit. b regarding the purpose limitation, lit. c in case of data minimisation is relevant for the design pattern. Here, opening clauses such as Art. 6 para. 3 of the GDPR and member state regulations based on this in the BDSG, HDSIG, and HHG may also have to be observed, particularly for data processing by public bodies. The design pattern suggests an emotion recognition on the device through an emotion ontology and provides the developer with

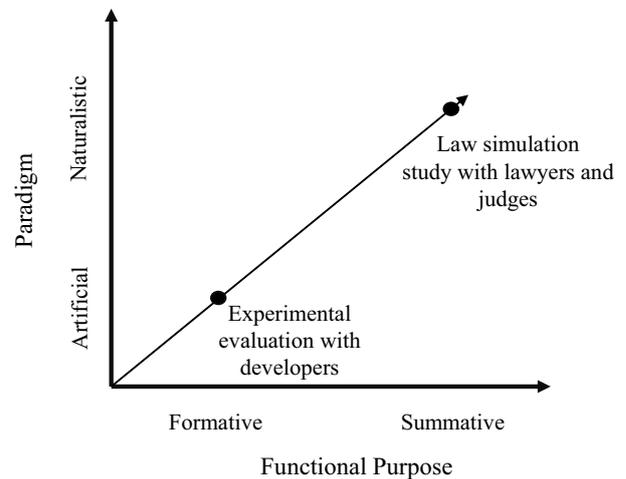


Figure 5. Classifying our evaluation approaches.

details to implement the patterns’ solution (see Figure 10 for the final design pattern).

During the workshops, eleven design patterns were developed: 1) processing emotional data, 2) privacy-friendly user profile, 3) profiling on foreign devices, 4) authorisation management, 5) deletion routines, 6) integration of external payment data, 7) data transfer to external devices, 8) sensitivity to wake words, 9) information assessment, 10) private mode, and 11) individual assistance. In the following, we evaluate to what extent the developed design patterns (and the underlying design pattern framework) support developers in developing lawful IT artefacts.

3.5. Evaluation of the design patterns

For our evaluation (see Figure 5), we applied the framework for evaluating DSR and relied on a twofold approach (Venable et al., 2016). First, we evaluated whether the design patterns are suitable to support developers in the process of designing lawful AI-based assistants using a formative evaluation approach in an artificial evaluation setting. Second, we introduced the simulation study approach – an evaluation method from the law discipline that allows us to evaluate how the design patterns contribute to system lawfulness using a summative evaluation approach (Roßnagel & Schuldt, 2013). By this means, we shed light on how the design patterns contribute to lawful system design. In the summative evaluation, developers used the design patterns to develop a lawful AI-based assistant. Afterwards, the AI-based assistant was used in a university lecture as exam preparation, and potentially arising conflicts in this setting were then evaluated in simulated court cases with real lawyers and judges. Therefore, we can evaluate the design patterns in a natural evaluation setting

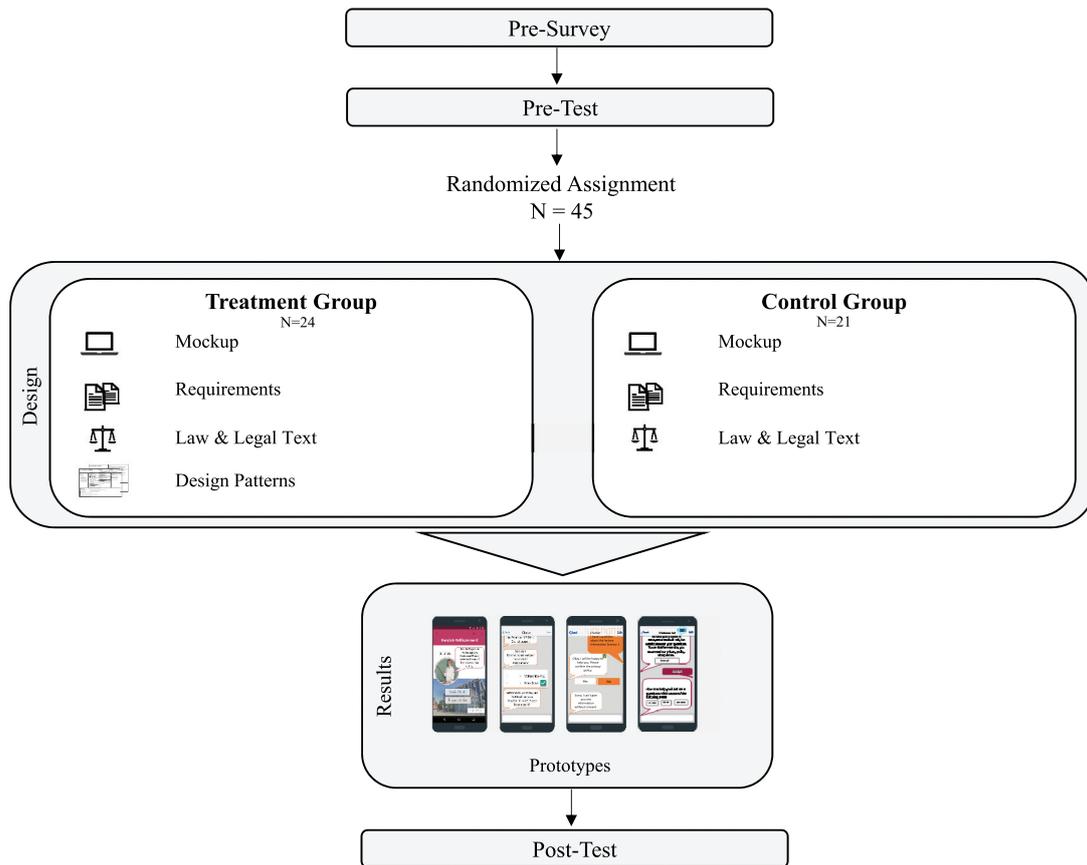


Figure 6. Formative evaluation process.

involving the development process of an IT artefact, the actual usage processes, and possible legal conflicts.

3.5.1. Formative evaluation episode from the developers' perspective

The design patterns were evaluated in a user study in two IS courses at a European university in a fully randomised between-subject experiment. In the evaluation, graduate students who were trained in requirements engineering as well as systems design applied the design patterns. Thus, the student population is a suitable sample to draw generalisable conclusions for evaluating the design pattern's utility in developing system prototypes when comparing the sample to IS developers that also never have been trained to explicitly design lawful IT artefacts (Compeau et al., 2012).

The task was to design a prototype of a chatbot learning assistant application. In total, 45 participants – all studying business with a major in IS – participated in the voluntary task, which was incentivised with three additional bonus points to improve their final grade. To ensure their seriousness in participating, extra credit could only be obtained when individuals followed the tasks as prescribed.

The participants were randomly divided into two groups. Concerning the experimental evaluation, we

included an experimental manipulation that corresponded to the provision of design patterns ($n = 24$). The experiment also included a control group ($n = 21$) without the support of design patterns (see Figure 6). All subjects were around the same age, ranging from 21 to 26 ($\bar{x} = 4.34$).

The evaluation task was to develop a smart learning assistant that supports the lecturer in answering questions and doing organisational tasks. Since the learning assistant would come into contact with personal or sensitive data such as grades, legal requirements and compliance with the GDPR are indispensable. All participants were provided with a mock-up framework as well as information regarding the use case, including interview documents of the lecturers for which the system is to be developed and relevant legal texts. In addition, we provided the experimental group with a design pattern catalogue besides the legal texts with the eleven design patterns. The participants in the experimental group were free to choose whether to use the patterns or not, but all participants decided to use the patterns. Afterwards, all participants answered a questionnaire, which allows us to receive feedback regarding the design pattern's usefulness and applicability.

The developed prototypes were evaluated by an expert panel assessment (one example prototype is presented in "Appendix B"). All experts have scientific

Table 4. Comparison between control and treatment groups' prototypes.

Expert Rating	Group	Support	Groups	Rating (1–5)	p-Value
	Control	No Support	N 21	Mean (SD) 2.1 (1.09)	P<.05
	Pattern	Design patterns (DP)	24	2.96 (1.41)	

publications either in the field of law ($n = 4$) or systems development ($n = 5$) but are not familiar with the design patterns and or the prototypes' group classification. The order in which the prototypes are presented is completely randomised.

We conducted the evaluation with two goals in mind: first, we investigate whether the design patterns support developers in the (cognitive) process of designing lawful AI-based assistants. Second, we examine from an outcome perspective whether the application of the design patterns result in lawful systems.² The results (see Table 4) indicate that the subjects who used the design patterns obtained significantly better ratings regarding the implementation of legal requirements in the expert evaluation. We use the feedback on the design pattern structure in the next section to revise the design pattern framework and the design patterns.

3.5.2. First revision of the design pattern framework

The evaluation indicated that the design patterns lead to the development of more lawful IT artefacts. Furthermore, the evaluation yielded qualitative feedback regarding the cognitive processes of the pattern deployment that we used to revise the design pattern framework. In addition, we derive one further pattern element resulting from the first evaluation episode. The fundamental structure of the design pattern has proven helpful in the evaluation, while the arrangement of the individual elements needs revision (see Figure 7). Now the goal and the requirements are in the upper design pattern area, and the solution is in the lower design pattern area. This supports users in reading the necessary information before presenting solutions. In addition to the illustration of the pattern, a new component has been added.

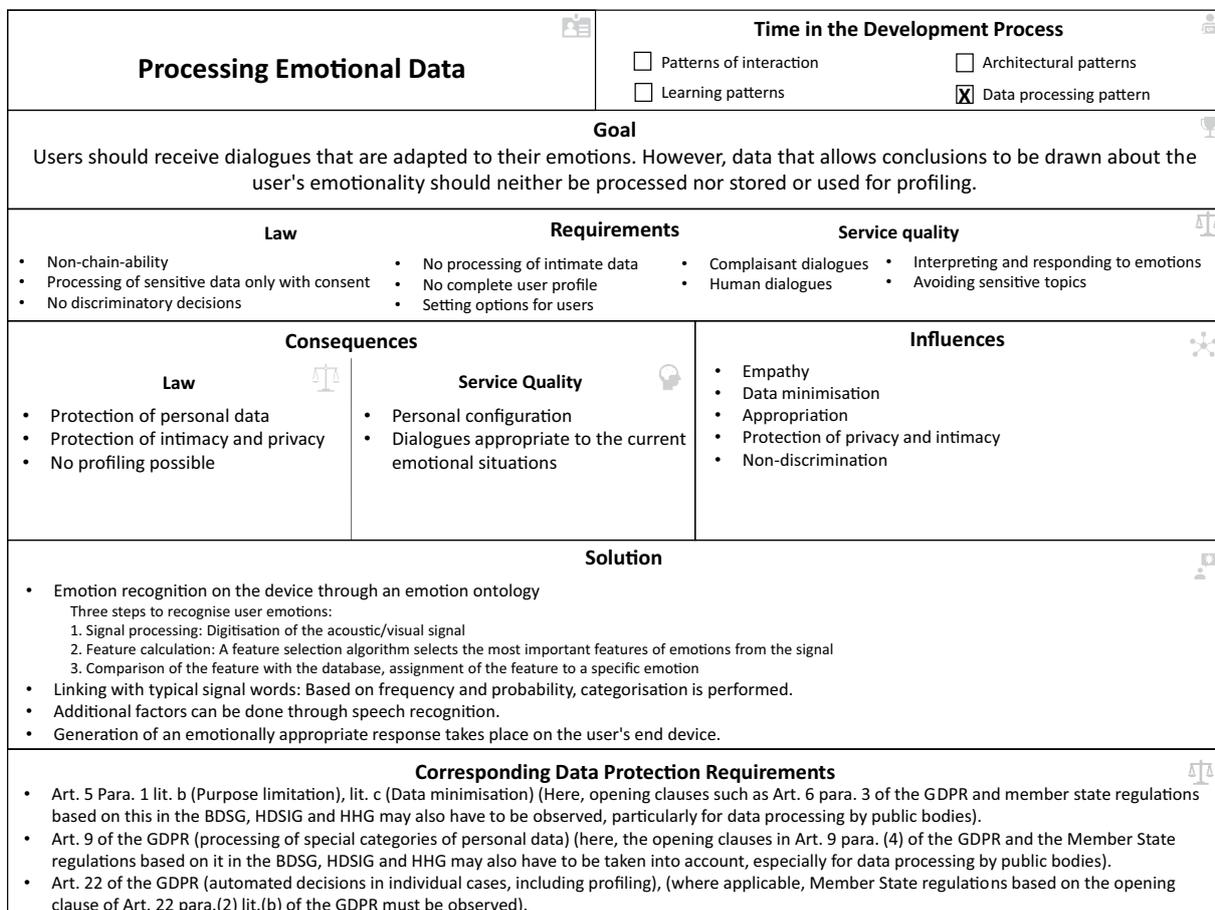


Figure 7. Design pattern after first revision.

The greatest challenge of the design pattern application was the linking of requirements and design patterns. The subjects in the formative evaluation spent a lot of time mapping requirements and design patterns. To integrate regulation rules into the artefact even better, we established the category “corresponding data protection requirements”. This category refers to concrete specifications that are anchored in the GDPR, for example, and enable the developer to draw better links between the law and the development. By using legal terminology, the developer can justify which laws they reference. Legal requirements are a core component of lawful systems development (Hoffmann et al., 2015) and are therefore one of the most important components of the design pattern. Therefore, we have integrated this as a core revision in the first revision. Hence, we propose our seventh pattern element:

Pattern Element #7: Integrate domain-specific information that helps users link practical implementation with abstract requirements.

Based on the new pattern element, we include a new category in our design pattern framework. “Corresponding data protection requirements” includes relevant information from the legal discipline. In addition, the new arrangement of the framework is clear and supports the reading direction from top to bottom. In the next step, we evaluate the revised patterns using a methodology from the discipline of law.

3.5.3. Summative evaluation using a legal simulation study

Simulation studies are an established method among legal experts to evaluate IT artefacts practically regarding their lawfulness in simulated court cases under real conditions (Roßnagel & Schuldt, 2013). Thus, a simulation study allows us to assess the developed design patterns in court cases. A key characteristic is that it facilitates realistic conditions while preventing damage. Therefore, it is desirable to provoke critical

situations (Pordesch et al., 1999). Using a simulation study, we can decide about how our developed design patterns contribute to developing lawful systems.

We received documents for analysis from four court cases. Additionally, we interviewed the judges and lawyers to obtain feedback on the patterns. The group interview was held with all participants ($N = 6$) after the simulation study and allowed the participants to exchange views on the patterns’ use as well as to extract and discuss critical aspects regarding revision of the patterns. The data (see Table 5) include two transcripts of the oral proceedings, the related correspondence between the lawyers and the judge before both oral proceedings, the documents from both written proceedings, and the interviews with the lawyers and judges after the hearings.

We use the legal simulation study to evaluate how the design patterns lead to the development of lawful artefacts (see Figure 8). The primary goal of the simulation study’s first part is to evoke legal disputes that serve as the basis for the court cases. This procedure enables the generation of legal violations as they might occur in reality by using the artefact in practice. For the user evaluation, we used the design patterns to develop a lawful AI-based assistant that is used as support in exam preparation as part of a course. We allowed students from a basic economics and business administration course to use the conversational assistant for half an hour per day for one week to review course material before the final assessment. The AI-based assistant, according to Knote et al. (2021), is a voice-based conversational agent for exam preparation in university courses. The teaching material is presented as a flashcard quiz to be as comprehensible as possible. The course content is repeated with individual users through multiple-choice questions and consolidated through gamified elements.

Based on the user evaluation, we negotiated the derived legal violations in the second part. The legal

Table 5. Documents collected from simulation study.

Relevant Documents from Simulation Study	Source	Scope
Filing of action of the case “passing personal data”	Exchange of pleadings	7 pages
Filing of the action of the case “omission of the use of the AI-based assistant”	Exchange of pleadings	4 pages
Statement of defence of the case “passing personal data”	Written hearing	5 pages
Statement of defence of the case “omission of the use of the AI-based assistant”	Written hearing	5 pages
Judgement of the case “passing personal data”	Written hearing	2 pages
Judgement of the case “omission of the use of the AI-based assistant”	Written hearing	3 pages
Video recording of case “storage duration of personal data”	Oral hearing	58 minutes
Video recording of case “right of data cancellation”	Oral hearing	46 minutes
Filing of action of the case “storage duration of personal data”	Exchange of pleadings	7 pages
Filing of action of the case “right of data cancellation”	Exchange of pleadings	6 pages
Statement of defence of the case “storage duration of personal data”	Oral hearing	5 pages
Statement of defence of the case “right of data cancellation”	Oral hearing	4 pages
E-learning charter of the university	Oral hearing	2 pages
Transcript of the case “storage duration of personal data”	Oral hearing	10 pages
Transcript of the case “right of data cancellation”	Oral hearing	8 pages
Interview transcript	Interview	5 pages

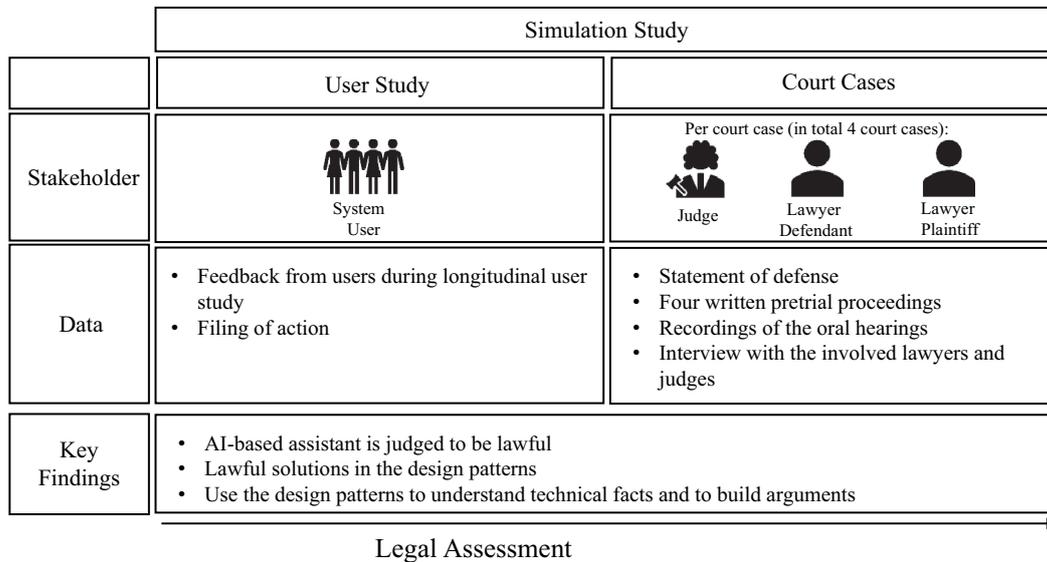


Figure 8. Legal simulation study.

violations in our simulation study follow the procedures of court proceedings according to German law. First, facts are ascertained through written communication via mails. Because the facts are not clarified, an oral hearing is held in accordance with Art. 6 para. 1 clause 1 ECHR. Thus, we simulate four court cases with real lawyers and judges acting on behalf of their law firm. Two judges and four lawyers participated in our simulation study – which consists of four court cases. All participants (one female; five males) had several years’ of professional experience as judge or lawyer.³ Two cases were heard before a civil court and two before an administrative court. We conducted the civil cases in written form and the two administrative law processes orally. Each trial involved a judge, a defendant lawyer, a plaintiff lawyer, the plaintiff, and the defendant. As plaintiffs, we recruited voluntary participants from the first part of the simulation study to present the process as realistically as possible. Thus, the plaintiffs were students. In all four cases, the defendant was represented by the university that used the IT artefact in the lecture.

In the following, we describe the process of one action as an example for the others. The simulation study starts with a pre-trial hearing to exchange the action and first evidence. The pre-trial written procedure took place over a period of 2–3 weeks and consisted of the plaintiff’s presentation of the facts, the defendant’s statement, and the judge’s invitation to the hearing. Both parties presented their evidence. In the example case, the reason for the action was the collection of personal data beyond the purpose of processing as well as information regarding the duration and purpose of data storage. The written preliminary proceedings began with a seven-page written plea in which the plaintiff’s lawyer conveyed the facts of the case and reasons for the action and asked the

defendant to refrain from using the IT artefact in university teaching (for an insight into the received court documents, please see “Appendix C”). In a five-page defence statement, the defendant’s lawyer commented on the action, referring to the patterns used in developing the IT artefact. In his defence, the lawyer referred to the design pattern “information assessment”.

After both parties exchanged their facts and evidence, the judge invited them to an oral dispute hearing. To address questions regarding the IT artefact’s development, an expert who had been involved in the development of the AI-based assistant was also invited to court. The expert left the court before the hearing began and only joined to answer questions about the development. According to the administrative court rules, the oral proceedings began with the presentation of the essential evidence. The judge presented the facts of the case and discussed the reasons for action. After the plaintiff’s lawyer confirmed the facts and set out the grounds in greater detail, the two lawyers and the judge examined the facts. Both parties now had the opportunity to present their cases, and the judge was informed of the situation. In our particular case, the plaintiff’s lawyer argued that the results of his exam preparation with the learning assistant were used beyond their intended purpose and were stored for an unnecessarily long time. After some exchanges, the defence lawyer used the design pattern “deletion routines” to point out the automatic deletion of the data at the end of the semester. In the interview the lawyer notes, “Whenever I was at a loss with my arguments, I could find technical details of the programming in the design patterns and use them for my arguments”. From this point on, the basis of argumentation shifted, and it had to be proven primarily that the design patterns were actually used. For this purpose, the neutral expert was involved again and questioned about

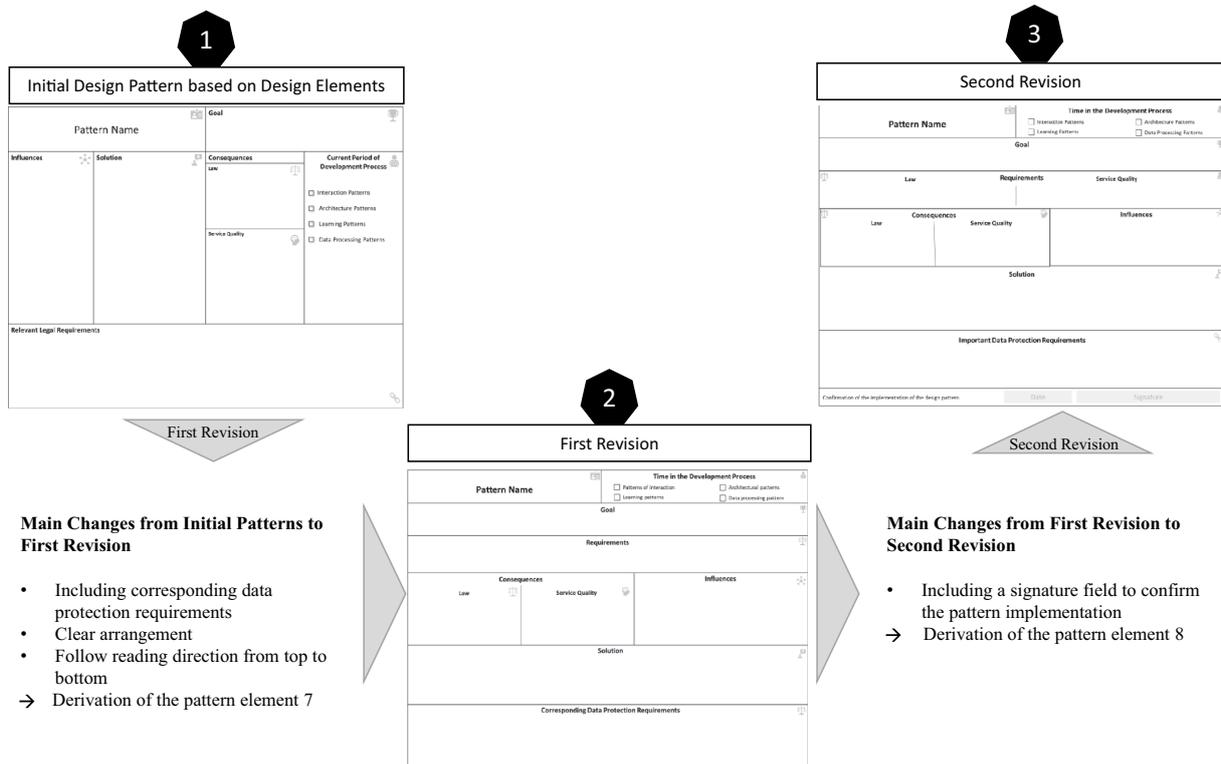


Figure 9. Design pattern revisions.

the development procedure. The negotiations ended with the pronouncement of a judgement. The judgement in our example action favoured the defendant, as the data storage in the AI-based assistant was judged to be lawful. The oral hearings lasted between 45 and 60 minutes.

The second evaluation episode examines the use of the design patterns in a natural and rather summative evaluation setting. Based on our analysis, we gain sufficient insights to draw conclusions about the lawfulness of the design pattern and the developed AI-based assistant. The AI-based assistant represents the design pattern and allows us to examine how the solution might be judged in actual court cases. All four judgements favoured the AI-based assistant developed using the design patterns. The results demonstrate two things: first, how the patterns are used in a holistic setting from development to legal evaluation of the developed artefact. Second, they show that the AI-based assistant and the solutions in the design patterns were judged to be lawful in all judgements. The evaluation yielded further insights for the pattern's revision (see Figure 9). We next used the evaluation to revise our design patterns.

3.5.4. Second revision of the design pattern framework

The law simulation study mainly provides evidence for two major aspects. First, the study provides deep insights into how design patterns

contribute to the lawfulness of an IT artefact. Second, the study highlights the utility of design patterns in court related to the process of assessing the degree of lawfulness during evidence exchanges and the arguments presented in court. However, in the oral actions, the challenge was to prove to the judge that the design pattern was actually implemented in the AI-based assistant. In the simulation study, it was found that the validity of the design pattern is related to the actual implementation. If the design patterns are to be used to assist legal experts in legal assessment or as a supplement to the technical documentation, it is necessary to confirm the actual implementation of the pattern by the developer. To circumvent this challenge, the eighth pattern element emerged. By providing the developer with a signature field to confirm the pattern's development, the legal experts no longer need to ask about the pattern's use. This saves time and contributes legal aspects to the development's documentation. Thus, we suggest our eighth pattern element:

Pattern element #8: Allow the developer to confirm the pattern's implementation.

We use the findings of the two evaluations to include the eighth pattern element in our design pattern framework (see Figure 9) and the design patterns for lawful AI-based assistants (see Figure 10).

Processing Emotional Data		Current Period of Development Process	
		<input type="checkbox"/> Patterns of interaction <input type="checkbox"/> Learning patterns	<input type="checkbox"/> Architectural patterns <input checked="" type="checkbox"/> Data processing pattern
Goal			
Users should receive dialogues that are adapted to their emotions. However, data that allows conclusions to be drawn about the user's emotionality should neither be processed nor stored or used for profiling.			
Law <ul style="list-style-type: none"> • Non-chain-ability • Processing of sensitive data only with consent • No discriminatory decisions 	Requirements <ul style="list-style-type: none"> • No processing of intimate data • No complete user profile • Setting options for users 	Service quality <ul style="list-style-type: none"> • Complaisant dialogues • Human dialogues 	<ul style="list-style-type: none"> • Interpreting and responding to emotions • Avoiding sensitive topics
Law <ul style="list-style-type: none"> • Protection of personal data • Protection of intimacy and privacy • No profiling possible 	Consequences <ul style="list-style-type: none"> • Personal configuration • Dialogues appropriate to the current emotional situations 	Influences <ul style="list-style-type: none"> • Empathy • Data minimisation • Appropriation • Protection of privacy and intimacy • Non-discrimination 	
Solution			
<ul style="list-style-type: none"> • Emotion recognition on the device through an emotion ontology Three steps to recognise user emotions: 1. Signal processing: Digitisation of the acoustic/visual signal 2. Feature calculation: A feature selection algorithm selects the most important features of emotions from the signal 3. Comparison of the feature with the database, assignment of the feature to a specific emotion • Linking with typical signal words: Based on frequency and probability, categorisation is performed. • Additional factors can be done through speech recognition. • Generation of an emotionally appropriate response takes place on the user's end device. 			
Important Data Protection Regulations			
<ul style="list-style-type: none"> • Art. 5 Para. 1 lit. b (Purpose limitation), lit. c (Data minimisation) (Here, opening clauses such as Art. 6 para. 3 of the GDPR and member state regulations based on this in the BDSG, HDSIG and HHG may also have to be observed, particularly for data processing by public bodies). • Art. 9 of the GDPR (processing of special categories of personal data) (here, the opening clauses in Art. 9 para. (4) of the GDPR and the Member State regulations based on it in the BDSG, HDSIG and HHG may also have to be taken into account, especially for data processing by public bodies). • Art. 22 of the GDPR (automated decisions in individual cases, including profiling), (where applicable, Member State regulations based on the opening clause of Art. 22 para.(2) lit.(b) of the GDPR must be observed). 			
Confirmation of implementation of contents of design pattern		Date	Signature

Figure 10. Final design pattern.

4. Discussion and implications

Our paper's goal was twofold. First, we aimed to contribute to codifying legal design knowledge through a design pattern framework by deriving pattern elements to develop useful design patterns. Second, we examined the extent to which design patterns may support developers to proactively consider IT regulation early on in systems development through the framework application in one specific case, i.e., the development of lawful AI-based assistants.

4.1. Discussion of findings

Considering our first research question, we investigated how we should codify legal design knowledge in design patterns to make the design knowledge as useful as possible. We used existing literature to develop a design pattern framework – not only literature on design patterns but also extant research from interdisciplinary research, practical insights from developers and lawyers, and a multistage evaluation including two revisions on codifying and presenting design knowledge. Design patterns have been established in various disciplines, such as architecture (Alexander, 1977), HCI (Borchers, 2002, April), and

systems development (Gamma, 1995), in recent decades. The literature has demonstrated the importance of the field of design knowledge codification for IS research (see e.g., Seidel et al., 2018; Vom Brocke et al., 2020). We thus integrated various findings and pursued our goal of practical, useful, and applicable design patterns. First, we derived requirements for the codification and representation of design patterns and then translated them into design pattern elements. By utilising cognitive load as a kernel theory for the requirements derivation, we ensured that the design patterns were designed keeping their utility during problem-solving processes such as IS development in mind. Second, we applied the design pattern framework to develop design patterns that provide lawful design knowledge for designing lawful AI-based assistants. To the best of our knowledge, ours is the first study to design and evaluate design patterns to support lawful systems development using a multi-stage, design science research approach. Our study complements research on the challenges of existing studies related to decontextualising the design process (Chandra Kruse et al., 2022; Tuunanen et al., 2022). These challenges are especially prevalent if they are related to solving ill-structured problems during the design of complex systems involving multiple,

interdisciplinary stakeholders (Dickhaut et al., 2023), e.g., designing healthcare systems versus designing lawful AI-based assistants in the educational context. In our paper, we want to disentangle this particular tension between nomothetic and idiographic approaches to advance the scientific discussion related to design knowledge through what Baskerville et al. (2015) describe as design-science duality. As mentioned earlier, design patterns are used in a variety of disciplines, but are often more practice oriented. Our framework is generally developed specifically for legal design knowledge but can be used and extended in certain parts for other disciplines as well. It is important that the specifics of the design knowledge of the respective discipline are taken into account when adapting the framework. On this basis we note that certain pattern elements of the framework might be well applicable to other domains. For example, the pattern element #3 with its focus on synthesising complex legal knowledge into information units could inform design patterns of other domains where complex knowledge is encapsulated into domain-specific knowledge sources. When considering the healthcare discipline where are also significant regulation and approval processes are involved, domain-specific and complex medical knowledge for designing systems could be in this case made more accessible for designers of IT artefacts.

Regarding our second research question, we applied the design pattern framework in one specific use case and evaluated to what degree the application of design patterns does lead to the development of lawful systems. First, we evaluated the design patterns with several subjects regarding practical applicability in an evaluation setting. The results indicated that the design patterns supported the subjects in developing significantly better-rated lawful prototypes. Going back to the kernel theory, we utilised our results to strengthen the argument that the principle of cognitive load is important to consider when taking IS development processes into account, e.g., by avoiding extraneous load when being confronted with two knowledge streams (in our case legal knowledge and knowledge related to systems).

Second, we reinforced our findings using a summative naturalistic evaluation. The simulation study allowed us to derive two findings: first, the confirmation that our design patterns' solutions are lawful, and second, the extent to which the design patterns promote lawful systems development by considering IT regulation rules early on in systems development. The four court judgements indicate that the AI-based assistant is judged to be lawful. In court, the design patterns were presented as evidence and played an important role. During the trial, the design patterns formed a basis for the lawyers to question their technical details and clarify the rationale for

implementation. Ultimately, the law simulation study strengthened the assumption that the first evaluation has already shown, namely that design patterns support the integration of legal IT regulation early on in systems development and thus promote the development of lawful IT artefacts.

The simulation study revealed a benefit of design patterns additionally to finding solutions to recurring legal issues during development: design patterns can serve as additional technical documentation and justification for decisions made in the design process, which might become relevant in related lawsuits or for compliance issues. In the oral court cases, the lawyers used the design patterns as evidence for decision-making during development. Analysis of the interview with the lawyers and judges revealed that design patterns may function as evidence in court. Using the simulation study, we have the special situation that it allows us to involve legal experts in our IS evaluation process, since they have the necessary knowledge to decide about how our developed design patterns contribute to system lawfulness.

4.2. Theoretical contributions and practical implications

We contribute to IS research in general and the practice of systems development by providing an approach that makes abstract regulation rules applicable for developers. Law is generally technology-neutral and must be considered individually in each specific application, making it difficult to integrate law with inadequate legal understanding (Hildebrandt & Tielemans, 2013). Considering the practical problem of equivocal regulation rules that Weick (2010) raised, the developed design patterns transform the abstract rules by making them implicit and more decisively applicable for developers. According to Smith and McKeen (2006), the responsibility for all regulations tend to fall at least partially on the shoulders of the developer. Therefore, our design pattern approach offers an adequate solution to these issues, especially given the outreach of the GDPR (Peukert et al., 2022), thus underscoring the importance of our approach. As Acquisti et al. (2022) state, a key challenge nowadays is to “embed privacy by default into the fabric of our digital system”, and design patterns for lawful systems development could be a valuable cornerstone in achieving this important goal.

Design patterns are based on the principle of providing proven solutions to recurring problems. In European courts, settlement proceedings are also often used in negotiating outstanding actions. This means that similar cases are taken at hand and argued based on their verdict. Thus, design patterns can support developers in finding adequate design solutions. In this context, our paper also contributes to the ongoing

discussion on how to effectively codify and accumulate design knowledge of design science research projects (Legner et al., 2020; Tuunanen & Holmström, 2021; Vom Brocke et al., 2020) and adds another angle to the debate with a design pattern-based view that is predominantly concerned with rather abstract design principles (Gregor et al., 2020). Thus, our design pattern-based approach contributes to DSR by providing an organising device to incrementally accumulate design knowledge according to design theories in general (Tuunanen & Holmström, 2021) and for considering legal requirements in specific.

Connected to the accumulative design knowledge aspect, practical implications are also related to the orchestration of software development through design patterns in use. Specifically, when drawing upon the call of Maruping and Matook (2020), integrating community actors to utilise and contribute to the presented legal design patterns in a platform-based repository could shed a light on the evolving nature of design patterns, especially related to the solution part of design patterns by involving a possibly large community of interdisciplinary actors. By this means, design patterns are, from a practical perspective, a device for creative solutions that tackle new and recurrent legal challenges when developing IT artefacts. Further, repositories could be the platform-based interface for the orchestration of these design efforts, e.g., through involving individuals from the legal but also computer science discipline, and could contribute to the practices of design knowledge accumulation and projectability through evolving and constantly renewing design patterns for legal aspects.⁴ In the end, this could also practically contribute to making design knowledge accessible for organisations with varying degrees of size and maturity related to covering legal aspects of IT development.

5. Limitations and future research

Despite our theoretical and practical contribution, our study has several limitations, which we would like to highlight with the need for further research. In our DSR approach (Pefferers et al., 2007), we demonstrate the feasibility and advantages of the developed design pattern framework applying to one specific instance: AI-based assistants. Our study shows what design patterns for the development of lawful AI-based assistants can look like. The developed patterns are designed for the development of lawful AI-based assistants. These may be, for example, voice assistants or chatbots. However, we have made the conscious decision not to create lawful design patterns for the development of all IT artefacts. To a certain extent, the solutions are transferable to other artefacts but would lose quality at a higher level of abstraction. Similarly, design patterns that relate exclusively to voice assistants would lead to limitations in the long term. The systems undergo constant

development, for example, voice assistants may be provided with a display and may no longer be purely voice-based (Schmitt et al., 2021). However, our contribution is not limited to AI-based assistants. Consideration of the GDPR is crucial for the development of many IT artefacts. Hence, future research should regard our research as a starting point and use the pattern elements and design pattern structure to further develop design patterns that integrate regulation in other research contexts and legal systems. In this context, large-scale field evaluations of patterns in use could also contribute to further the discussion about legal design knowledge.

We evaluated our design patterns with two evaluation episodes. First, we acknowledge that the evaluation with student developers faces limitations in terms of generalisability. Although we purposefully designed the experiment with a complementing simulation study in mind, future research should take our design patterns as a resource to experiment with different populations, e.g., related to prior experience with legal design of IT artefacts. In that context, we also note that additional experiments in the field and (subjective) cognitive load measurements could contribute to a further exploration on how design patterns contribute to the efficiency of developing lawful IT artefacts. Second, we evaluated our design patterns under German law through simulation study court hearings. German law is among the strictest with respect to data protection (Li et al., 2019) and is therefore also relevant for other countries. We note in terms of generalisability that the GDPR covers the most extensive set of individual rights related to data protection and privacy, while other regulations such as the prominent California Consumer Privacy Act (CCPA) also share a majority of principles with the GDPR (Aljeraisy et al., 2021). Nonetheless, future research should also apply the simulation study to other legal systems to extend our approach and provide insights on its generalisability for evaluating privacy-related artefacts in IS research.

The use of the design patterns is intended to support developers in the development of lawful IT artefacts. Basically, it always requires a great deal of trust. However, the implementation of the design pattern is not a guarantee of legality. Even if a design pattern has been fully implemented, other factors are relevant in the legal assessment. If we consider the results of Vaujany et al. (2018), we classify the design patterns as a materialisation between rule and IT artefact in which the design pattern supports this stage. Due to the technology neutrality of the law, their implementation remains to a certain extent a matter of “interpretation” and must be considered specifically for each individual case. However, the design patterns show proven solutions and thus support the application of best practices for complicated legal requirements. In that context, we also highlight that

applying the framework of design patterns to domains could be a fruitful way for DSR to engage with the debate on accumulating and communicating design knowledge beyond mere design principles (Chandra Kruse et al., 2022; Tuunanen & Holmström, 2021).

6. Conclusion

In this paper, we provide a design pattern framework to the consideration of regulations early in the systems development process. We derived design pattern requirements based on literature and a practitioners' focus group workshop on codifying design knowledge. Based on the requirements, we generated pattern elements applicable to the development of design patterns for legal knowledge. Our results show that the codification of legal knowledge in design patterns yields better results regarding the development of lawful IT artefacts. Therefore, we provide pattern elements and a design pattern framework to codify (legal) design knowledge. We demonstrate the feasibility and advantages of these design patterns for lawful IS using the example of AI-based assistants.

The results show how early consideration of regulation rules can help to develop sustainable, lawful IT. With our contribution, we have both given a theoretical and a practical contribution for the codification of design knowledge in design patterns and have extracted a pattern structure through multiple and rich evaluations, as well as pattern elements that can support other researchers in codifying design knowledge through design patterns.

Notes

1. We note that in its original triarchic formulation, CLT also considers germane load. Nonetheless, we build upon the revised CLT conceptualisation that highlights that germane load is essentially indistinguishable from intrinsic load, and therefore this third load type may be redundant. This is in line with the goal of the paper, which is to seek to manage overall cognitive load for better task performance.
2. As Paas & van Gog (2006) note, when considering the effects of cognitive load management, it is of utmost importance to consider an outcome perspective related to cognitive processes that we present with the assessment of the developed prototypes. Thus, we refrained from measuring subjective cognitive load measures that are subject to significant criticism but rather considered qualitative feedback from the subjects to inform the first design pattern framework revision.
3. In addition, all participants had completed the second German state examination in law.
4. The developed design patterns are accessible as supplementary material in the online appendix and are

licensed under a CC-BY license to allow contributions to and customizing of the design patterns.

5. Dickhaut, E., Janson, A., & Leimeister, J. M. 2020. Codifying Interdisciplinary Design Knowledge Through Patterns - The Case of Smart Personal Assistants. In S. Hofmann, O. Müller & M. Rossi (Eds.), *Designing for Digital Transformation. Co-Creating Services with Citizens and Industry*: 114–125. Cham: Springer International Publishing.

Acknowledgements

This paper presents research that was conducted in context of the project “AnEkA” (project number: 348084924), funded by the German Research Foundation (DFG). The second author acknowledges funding by the basic research fund from the University of St. Gallen. Furthermore, this research builds on a paper that has been presented at the International Conference on Design Science Research in Information Systems and Technology (DESRIST) 2020.⁵ We thank the reviewers and attendees for their valuable feedback that helped us to improve our research and to write this paper. Last but not least, we thank the editors for the guidance as well as the anonymous reviewers for their constructive feedback during the review process.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This paper presents research that was conducted in context of the project AnEkA (project 348084924), funded by the German Research Foundation (DFG).

ORCID

Ernestine Dickhaut  <http://orcid.org/0000-0002-1946-4038>

Andreas Janson  <http://orcid.org/0000-0003-3149-0340>

Matthias Söllner  <http://orcid.org/0000-0002-1347-8252>

Jan Marco Leimeister  <http://orcid.org/0000-0002-1990-2894>

References

- Acquisti, A., Brandimarte, L., & Hancock, J. (2022). How privacy's past may shape its future. *Science (New York, NY)*, 375(6578), 270–272. <https://doi.org/10.1126/science.abj0826>
- Ahrens, J., & Sankar, C. (1993). Tailoring database training for end users. *MIS Quarterly*, 17(4), 419–439. <https://doi.org/10.2307/249586>
- Alexander, C. (1977). *A pattern language: Towns, buildings, construction*. Oxford University Press.
- Alexander, C. (1979). *The timeless way of building* (24. print). *Center for Environmental Structure series*. Oxford University Press.
- Aljerais, A., Barati, M., Rana, O., & Perera, C. (2021). Privacy laws and privacy by design schemes for the

- internet of things. *ACM Computing Surveys*, 54(5), 1–38. <https://doi.org/10.1145/3450965>
- Almeida, P. G. R. D., Denner dos Santos, C., & Silva Farias, J. (2020). Artificial intelligence regulation: A meta-framework for formulation and governance. *Proceedings of the 53rd Hawaii International Conference on System Sciences*, Maui.
- Avdiji, H., Elikan, D., Missonier, S., & Pigneur, Y. (2020). A design theory for visual inquiry tools. *Journal of the Association for Information Systems*, 21(3), 695–734. <https://doi.org/10.17705/1jais.00617>
- Ayala-Rivera, V., & Pasquale, L. (2018). The grace period has ended: An approach to operationalize GDPR requirements. *Proceedings - 2018 IEEE 26th International Requirements Engineering Conference*, 136–146. <https://doi.org/10.1109/re.2018.00023>
- Ayres, P., & Sweller, J. (2005). The split-attention principle in multimedia learning. *The Cambridge Handbook of Multimedia Learning*, 2, 135–146.
- Barati, M., Petri, I., & Rana, O. F. (2019). Developing GDPR compliant user data policies for internet of things. *Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing*, Auckland, New Zealand, 133–141.
- Baruh, L., Secinti, E., & Cemalcilar, Z. (2017). Online privacy concerns and privacy management: A meta-analytical review. *The Journal of Communication*, 67(1), 26–53. <https://doi.org/10.1111/jcom.12276>
- Baskerville, R., Kaul, M., & Storey, V. C. (2015). Genres of inquiry in design-science research: justification and evaluation of knowledge production. *MIS Quarterly*, 39(3), 541–564. <https://doi.org/10.25300/MISQ/2015/39.3.02>
- Baxter, D., Gao, J., Case, K., Harding, J., Young, B., Cochrane, S., & Dani, S. (2007). An engineering design knowledge reuse methodology using process modelling. *Research in Engineering Design*, 18(1), 37–48. <https://doi.org/10.1007/s00163-007-0028-8>
- Becker, J., Heddier, M., Braeuer, S., & Knackstedt, R. (2014). Integrating regulatory requirements into information systems design and implementation. *Thirty Fifth International Conference on Information Systems*, Auckland 2014.
- Beverungen, D., Müller, O., Matzner, M., Mendling, J., & Vom Brocke, J. (2019). Conceptualizing smart service systems. *Electronic Markets*, 29(1), 7–18. <https://doi.org/10.1007/s12525-017-0270-5>
- Blind, K., Petersen, S. S., & Riillo, C. A. F. (2017). The impact of standards and regulation on innovation in uncertain markets. *Research Policy*, 46(1), 249–264. <https://doi.org/10.1016/j.respol.2016.11.003>
- Borchers, J. (2002, April). Teaching HCI design patterns: Experience from two university courses. *Patterns in practice: A workshop for UI designers (at CHI 2002 international conference on human factors of computing systems)*.
- Büthe, T., & Mattli, W. (2013). *The new global rulers: The privatization of regulation in the world economy* (3. pr., 1. pbk. pr). Princeton Univ. Press.
- Butler, T. (2017). Towards a standards-based technology architecture for RegTech. *Journal of Financial Transformation*, 45(1), 49–59. <https://files.openpdfs.org/jel1d4gbz5ob.pdf#page=49>
- Chandra Kruse, L., & Nickerson, J. V. (2018). Portraying design essence. *HICSS*, 4433–4442. <https://doi.org/10.2139/ssrn.3039322>
- Chandra Kruse, L., Purao, S., & Seidel, S. (2022). How designers use design principles: design behaviors and application modes. *Journal of the Association for Information Systems (JAIS)*, 23(5), 1235–1270. <https://doi.org/10.17705/1jais.00759>
- Chandra Kruse, L., & Seidel, S. (2017). Tensions in design principle formulation and reuse. *Designing the Digital Transformation DESRIST Research in Progress Proceedings of the 12th International Conference on Design Science Research in Information Systems and Technology*, Karlsruhe, Germany, 180–188.
- Chandra, L., Seidel, S., & Gregor, S. (2014). Prescriptive knowledge in is research: Conceptualizing design principles in terms of materiality, action, and boundary conditions. *HICSS*, 4039–4048. <https://doi.org/10.1109/HICSS.2015.485>
- CMS Legal. (2021). GDPR Enforcement Tracker. Retrieved January 19, 2021.
- Compagna, L., El Khoury, P., Krausová, A., Massacci, F., & Zannone, N. (2009). How to integrate legal requirements into a requirements engineering methodology for the development of security and privacy patterns. *Artificial Intelligence and Law*, 17(1), 1–30. <https://doi.org/10.1007/s10506-008-9067-3>
- Compeau, D., Marcolin, B., Kelley, H., & Higgins, C. (2012). Research Commentary—Generalizability of Information Systems Research Using Student Subjects—A Reflection on Our Practices and Recommendations for Future Research. *Information Systems Research*.
- Cowan, B. R., Pantidi, N., Coyle, D., Morrissey, K., Clarke, P., Al-Shehri, S., Earley, D., & Bandeira, N. (2017). “What can i help you with?”: Infrequent users’ experiences of intelligent personal assistants. *Association for Computing Machinery*, 1–12. <https://doi.org/10.1145/3098279.3098539>
- Dickhaut, E., Janson, A., Hevner, A. R., & Leimeister, J. M. (2023). Sharing design knowledge through codification in interdisciplinary DSR collaborations, Maui, Hawaii.
- Dickhaut, E., Janson, A., & Leimeister, J. M. (2020). Codifying interdisciplinary design knowledge through patterns—the case of smart personal assistants. *In 15th International Conference on Design Science Research in Information Systems and Technology, DESRIST 2020*, Kristiansand, Norway, December 2–4, 2020, Proceedings 15 (pp. 114–125). Springer International Publishing.
- Doering, A., & Veletsianos, G. (2007). Multi-Scaffolding environment: An analysis of scaffolding and its impact on cognitive load and problem-solving ability. *Journal of Educational Computing Research*, 37(2), 107–129. <https://doi.org/10.2190/Q58T-4388-8015-8141>
- Elshan, E., Engel, C., Ebel, P., & Siemon, D. (2022). Assessing the reusability of design principles in the realm of conversational agents. *DESRIST*, 13229, 128–141. https://doi.org/10.1007/978-3-031-06516-3_10
- European Data Protection Board. (2021). *Guidelines 02/2021 on Virtual Voice Assistants*. https://edpb.europa.eu/our-work-tools/public-consultations-art-704/2021/guidelines-022021-virtual-voice-assistants_de
- European Union. (2018). *General data protection regulation: (GDPR)*.
- Figl, K. (2017). Comprehension of procedural visual business process models - a literature review. *Business & Information Systems Engineering*, 59(1), 41–67. <https://doi.org/10.1007/s12599-016-0460-2>
- Ford, A., Al-Nemrat, A., Ghorashi, S. A., & Davidson, J. (2021). The impact of GDPR infringement fines on the market value of firms. *ECCWS 2021-Proceeding of the 20th European Conference on Cyber Warfare and*

- Security*, 24–25. Academic Conferences International Limited. <https://doi.org/10.34190/EWS.21.088>
- Furey, E., & Blue, J. (2018). Alexa, emotions, privacy and GDPR. *Proceedings of the 32nd International BCS Human Computer Interaction Conference 32*, 1–5. <https://doi.org/10.14236/ewic/HCI2018.212>
- Gamma, E. (1995). Design patterns: elements of reusable object-oriented software. Pearson Education India.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design Patterns: Elements of Reusable Object Oriented Software. AddisonWesley Professional.
- Garud, R. (1997). On the distinction between know-how, know-why, and know-what. *Advances in Strategic Management*, 14(14), 81–101.
- Geradin, D., Karanikioti, T., & Katsifis, D. (2021). Gdpr Myopia: How a well-intended regulation ended up favouring large online platforms - the case of ad tech. *European Competition Journal*, 17(1), 47–92. <https://doi.org/10.1080/17441056.2020.1848059>
- Gregor, S., Kruse, L., & Seidel, S. (2020). Research perspectives: The anatomy of a design principle. *Journal of the Association for Information Systems*, 21, 1622–1652. <https://doi.org/10.17705/1jais.00649>
- Guerra, K., & Koh, C. (2019). Do legal systems affect the organizational consequences of IT innovation? *Twenty-Fifth Americas Conference on Information Systems*, Munich, Germany, Cancun.
- Hadar, I., Hasson, T., Ayalon, O., Toch, E., Birnhack, M., Sherman, S., & Balissa, A. (2018). Privacy by designers: Software developers' privacy mindset. *Empirical Software Engineering*, 23(1), 259–289. <https://doi.org/10.1007/s10664-017-9517-1>
- Hauswald, J., Laurenzano, M. A., Zhang, Y., Li, C., Rovinski, A., Khurana, A., Dreslinski, R. G., Mudge, T., Petrucci, V., Tang, L., & Mars, J. (2015). Sirius: An open end-to-end voice and vision personal assistant and its implications for future warehouse scale computers. *Proceedings of the Twentieth International Conference on Architectural Support for Programming Languages and Operating Systems*, 223–238. <https://doi.org/10.1145/2694344.2694347>
- Heinrich, P., & Schwabe, G. (2014). Communicating nascent design theories on innovative information systems through multi-grounded design principles, Miami, FL, USA, 148–163.
- Hildebrandt, M., & Tielemans, L. (2013). Data protection by design and technology neutral law. *Computer Law & Security Review*, 29(5), 509–521. <https://doi.org/10.1016/j.clsr.2013.07.004>
- Hoffmann, A., Schulz, T., Zirfas, J., Hoffmann, H., Roßnagel, A., & Leimeister, J. M. (2015). Legal compatibility as a characteristic of sociotechnical systems. *Business & Information Systems Engineering*, 57(2), 103–113. <https://doi.org/10.1007/s12599-015-0373-5>
- Hsiao, I. -H., & Lopez, C. (2016). Lessons learned from students' cheat sheets: Generic models for designing programming study guides. *IEEE 16th International Conference on Advanced Learning Technologies (ICALT)*, Austin, TX, USA, 209–211.
- Huth, D., Both, A., Ahmad, J., Sauer, G., Yilmaz, F., & Matthes, F. (2020). Process and tool support for integration of privacy aspects in agile software engineering. *Americas Conference on Information Systems (AMCIS) Proceedings*, Utah, USA.
- Janson, A., Söllner, M., & Leimeister, J. M. (2020). Ladders for learning: Is scaffolding the key to teaching problem solving in technology-mediated learning contexts? *Academy of Management Learning & Education*, 19(4), 439–468. <https://doi.org/10.5465/amle.2018.0078>
- Kalyuga, S. (2011). Cognitive load theory: How many types of load does it really need? *Educational Psychology Review*, 23(1), 1–19. <https://doi.org/10.1007/s10648-010-9150-7>
- Kalyuga, S., Ayres, P., Chandler, P., & Sweller, J. (2009). The expertise reversal effect. *Educational Psychologist*, 38(1), 23–31. https://doi.org/10.1207/S15326985EP3801_4
- Kirschner, P. A., Ayres, P., & Chandler, P. (2011). Contemporary cognitive load theory research: The good, the bad and the ugly. *Computers in Human Behavior*, 27(1), 99–105. <https://doi.org/10.1016/j.chb.2010.06.025>
- Knackstedt, R., Heddier, M., & Becker, J. (2014). Conceptual modeling in law: An interdisciplinary research agenda. *Communications of the Association*, 34. <https://doi.org/10.17705/1CAIS.03436>
- Knote, R., Janson, A., Söllner, M., & Leimeister, J. M. (2021). Value co-creation in smart services: A functional affordances perspective on smart personal assistants. *Journal of the Association for Information Systems (JAIS)*, 22(2), 418–458. <https://doi.org/10.17705/1jais.00667>
- Koukoultsos, K., Khazaei, B., Dearden, A., & Ozcan, M. (2009). *Teaching usability principles with patterns and guidelines*. https://doi.org/10.1007/978-0-387-89022-7_11
- Laux, J., Wachter, S., & Mittelstadt, B. (2021). Taming the few: Platform regulation, independent audits, and the risks of capture created by the DMA and DSA. *Computer Law & Security Review*, 43, 105613. <https://doi.org/10.1016/j.clsr.2021.105613>
- Lau, J., Zimmerman, B., & Schaub, F. (2018). Alexa, are you listening? *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW), 1–31. <https://doi.org/10.1145/3274371>
- Legner, C., Pentek, T., & Otto, B. (2020). Accumulating design knowledge with reference models: Insights from 12 years' research into data management. *Journal of the Association for Information Systems (JAIS)*, 21(3), 735–770. <https://doi.org/10.17705/1jais.00618>
- Li, H., Yu, L., & He, W. (2019). The impact of GDPR on global technology development. *Journal of Global Information Technology Management*, 22(1), 1–6. <https://doi.org/10.1080/1097198X.2019.1569186>
- Lowry, P. B., Dinev, T., & Willison, R. (2017). Why security and privacy research lies at the centre of the information systems (IS) artefact: Proposing a bold research agenda. *European Journal of Information Systems*, 26(6), 546–563. <https://doi.org/10.1057/s41303-017-0066-x>
- Malinova, M., & Mendling, J. (2013). The effect of process map design quality on process management success. *ECIS*. https://aisel.aisnet.org/ecis2013_cr/160
- Martin, N., & Matt, C. (2018). Unblackboxing the effects of privacy regulation on startup innovation. *Thirty Ninth International Conference on Information Systems*, San Francisco, USA.
- Maruping, L. M., & Matook, S. (2020). The evolution of software development orchestration: Current state and an agenda for future research. *European Journal of Information Systems*, 29(5), 443–457. <https://doi.org/10.1080/0960085X.2020.1831834>
- McLeod, L., & MacDonell, S. G. (2011). Factors that affect software systems development project outcomes. *ACM Computing Surveys*, 43(4), 1–56. <https://doi.org/10.1145/1978802.1978803>
- Medina-Borja, A. (2015). Editorial Column—Smart things as service providers: A call for convergence of disciplines

- to build a research agenda for the service systems of the future. *Service Science*, 7(1), ii–v. <https://doi.org/10.1287/serv.2014.0090>.
- Nonaka, I., & Toyama, R. (2003). The knowledge-creating theory revisited: Knowledge creation as a synthesizing process. *Knowledge Management Research & Practice*, 1(1), 2–10. <https://doi.org/10.1057/palgrave.kmrp.8500001>
- Osterwalder, A., & Pigneur, Y. (2013). Designing business models and similar strategic objects: The Contribution of IS. *Journal of the Association for Information Systems*, 14(5), 237–244. <https://doi.org/10.17705/1jais.00333>
- Paas, F., & van Gog, T. (2006). Optimising worked example instruction: Different ways to increase germane cognitive load. *Learning and Instruction*, 16(2), 87–91. <https://doi.org/10.1016/j.learninstruc.2006.02.004>
- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
- Petit, N. (2021). The Proposed Digital Markets Act (DMA): A legal and policy review. *Journal of European Competition Law & Practice*, 12(7), 529–541. <https://doi.org/10.1093/jeclap/lpab062>
- Petter, S., Khazanchi, D., & Murphy, J. D. (2010). A design science based evaluation framework for patterns. *ACM SIGMIS Database: The DATABASE for Advances in Information Systems*, 41(3), 9–26. <https://doi.org/10.1145/1851175.1851177>
- Peukert, C., Bechtold, S., Batikas, M., & Kretschmer, T. (2022). Regulatory spillovers and data governance: Evidence from the GDPR. *Marketing Science*, 41(4), 746–768. Article mksc.2021.1339. Advance online publication. <https://doi.org/10.1287/mksc.2021.1339>
- Pordesch, V., Roßnagel, A., & Schneider, M. (1999). Simulation study mobile and secure communication in healthcare. *DuD*, 23(2), 76–80.
- PricewaterhouseCoopers. (2017). Pulse survey: US companies ramping up General Data Protection Regulation (GDPR) budgets. <https://www.Pwc.Com/us/en/press-Releases/2017/pwc-Gdpr-Compliance-Press-Release.Html>.
- Reiser, B. J. (2004). Scaffolding complex learning: The mechanisms of structuring and problematizing student work. *Journal of the Learning Sciences*, 13(3), 273–304. https://doi.org/10.1207/s15327809jls1303_2
- Roßnagel, A., & Schuldt, M. 2013. *The simulation study as a method of evaluating socially acceptable technology design*. Springer, Cham.
- Rothe, H., Wessel, L., & Barquet, A. P. (2020). Accumulating design knowledge: A mechanisms-based approach. *Journal of the Association for Information Systems*, 21(3), 771–810. <https://doi.org/10.17705/1jais.00619>
- Sagr, M. (2022). Is GDPR failing? a tale of the many challenges in interpretations, applications, and enforcement. *International Journal of Health Sciences*, 16(5), 1–2. <https://ijhs.org.sa/index.php/journal/article/download/7339/1118>
- Schmitt, A., Zierau, N., Janson, A., & Leimeister, J. M. (2021). Voice as a contemporary frontier of interaction design. *ECIS 2021 Proceedings*, Marrakech, Morocco.
- Schoormann, T., Möller, F., & Hansen, M. R. P. (2021). How do researchers (re-)use design principles: An inductive analysis of cumulative research, Kristiansand, Norway, 188–194.
- Security Week. (2020). Zoom’s security and privacy woes violated GDPR, expert says. <https://www.Securityweek.Com>.
- Seidel, S., Chandra Kruse, L., Székely, N., Gau, M., Stieger, D., Peffer, K., Tuunanen, T., Niehaves, B., & Lyytinen, K. (2018). Design principles for sensemaking support systems in environmental sustainability transformations. *European Journal of Information Systems*, 27(2), 221–247. <https://doi.org/10.1057/s41303-017-0039-0>
- Siena, A., Mylopoulos, J., Perini, A., & Susi, A. (2009). Designing law-compliant software requirements. *International Conference on Conceptual Modeling*, 472–486. https://doi.org/10.1007/978-3-642-04840-1_35
- Smith, H. A., & McKeen, J. D. 2006. Developments in Practice XXI: IT in the New World of Corporate Governance Reforms. *Communications of the Association for Information Systems*, 17. <https://doi.org/10.17705/1CAIS.01732>.
- Spiekermann, S. (2012). The challenges of privacy by design. *Communications of the ACM*, 55(7), 38–40. <https://doi.org/10.1145/2209249.2209263>
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive science*, 12(2), 257–285. https://doi.org/10.1207/s15516709cog1202_4
- Sweller, J., van Merriënboer, J. J. G., Paas, F., & C, G. W. (1998). Cognitive architecture and instructional design. *Educational Psychology Review*, 10(3), 251–296. <https://doi.org/10.1023/A:1022193728205>
- Taylor, P. R. (2001). Patterns as software design canon. *ACIS 2001 Proceedings*, 65.
- Tuunanen, T., & Holmström, J. (2021). *Incremental accumulation of information systems design theory*. https://doi.org/10.1007/978-3-030-84655-8_10
- Tuunanen, T., Salo, M., & Li, F. (2022). Modular service design of information technology-enabled services. *Journal of Service Research*, 109467052210827. <https://doi.org/10.1177/10946705221082775>
- van Aken, J. E. (2005). Valid knowledge for the professional design of large and complex design processes. *Design Studies*, 26(4), 379–404. <https://doi.org/10.1016/j.destud.2004.11.004>
- Vanberg, A. D. (2021). Informational privacy post GDPR – end of the road or the start of a long journey? *The International Journal of Human Rights*, 25(1), 52–78. <https://doi.org/10.1080/13642987.2020.1789109>
- van der Syde, Y. S., & Maalej, W. (2014). On lawful disclosure of personal user data: What should app developers do? *RELAW*, 25–34. <https://doi.org/10.1109/relaw.2014.6893479>
- Vaujany, F. -X.D., Fomin, V. V., Haefliger, S., & Lyytinen, K. (2018). Rules, practices, and information technology: A Trifecta of organizational regulation. *Information Systems Research*, 29(3), 755–773. <https://doi.org/10.1287/isre.2017.0771>
- Väyrynen, K., & Lanamäki, A. (2020). Policy ambiguity and regulative legitimacy of technology: Legal indeterminacy as result of an ambiguous taximeter regulation. *Forty-First International Conference on Information Systems*, Hyderabad, India, India.
- Venable, J., Pries-Heje, J., & Baskerville, R. (2016). FEDS: A framework for evaluation in design science research. *European Journal of Information Systems*, 25(1), 77–89. <https://doi.org/10.1057/ejis.2014.36>
- Vom Brocke, J., Winter, R., Hevner, A., & Maedche, A. (2020). Special issue editorial –accumulation and evolution of design knowledge in design science research: A journey through time and space. *Journal of the Association for Information Systems*, 23(3), 9–49. <https://doi.org/10.17705/1jais.00611>

- Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an information system design theory for vigilant EIS. *Information Systems Research*, 3(1), 36–59. <https://doi.org/10.1287/isre.3.1.36>
- Wang, Z., Sundin, L., Murray-Rust, D., & Bach, B. (2020). Cheat sheets for data visualization techniques, Atlanta, USA, 1–13.
- Wania, C. (2019). Exploring design patterns as evaluation tools in human computer interaction education. *MWAIS*, 9.
- Weick, K. E. (2010). *Sensemaking in organizations [Nachdr.] foundations for organizational science*. Sage Publ.
- Yskout, K., Scandariato, R., & Joosen, W. (2015). Do security patterns really help designers? *International Conference on Software Engineering*, Florence, Italia, 292–302.
- Zahedi, M., & Babar, M. A. (2014). Knowledge sharing for common understanding of technical specifications through artifactual culture. *EASE*. Advance online publication. <https://doi.org/10.1145/2601248.2601293>

Appendix A

Appendix A: Focus Group Participants

Table A1. Focus Group Participants.

Participant	Education	Work experience	Practical design pattern experience
#1	PhD in IS	IT consultant	yes
#2	PhD in Law	Lawyer	no
#3	Computer science student	Developer	no
#4	Computer science student	-	no
#5	PhD student in IS	UX consultant	yes
#6	Postdoctoral researcher in IS	Project manager	no
#7	Postdoctoral researcher in computer science	IT consultant	no
#8	Postdoctoral researcher in computer science	Developer	yes

Appendix B

Appendix B: Exemplary Developed Prototypes (Evaluation 1)

In our first evaluation we provide students with mock-ups to develop lawful AI-based assistant. The task was to develop a lawful learning assistant for higher education. The participants were completely randomly divided into two groups. We integrate an experimental manipulation that corresponded to the provision of design patterns (n=15). The experiment also included a control group (n=13) without the support of patterns.

To make the underlying conditions of the prototypes comparable, we gave the participants a mock-up consisting of a cell phone layout. Like the task description, the prototypes are formulated in German. In the following a screenshot of a prototype with support of the design patterns and a prototype without support of the design patterns is shown.

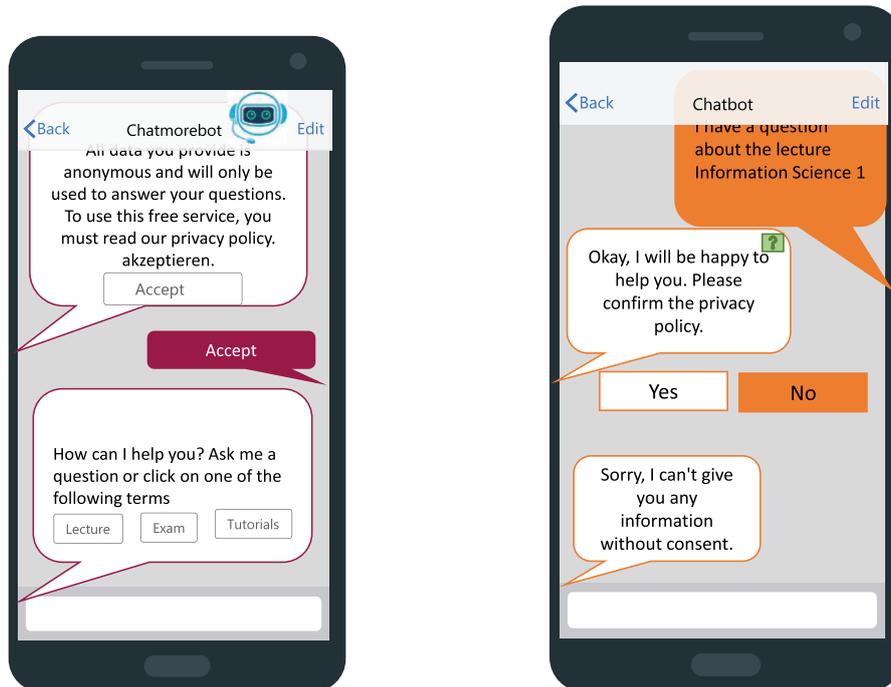


Figure B1. Exemplary Prototypes as Result from the Experiment. Left Prototype Shows Results from the Treatment Condition. Right Prototype Shows Results from the Control Condition (translated into English).

Appendix C

Appendix C: Exemplary Court Documents (Evaluation 2)

Within the simulation study we have received various documents. The documents reflect the realisation of the simulation study as real court cases. The documents are written in German, as is the trial itself. The following are excerpts from a trial: statement of claim, statement of defence and replication. We have anonymised the documents.

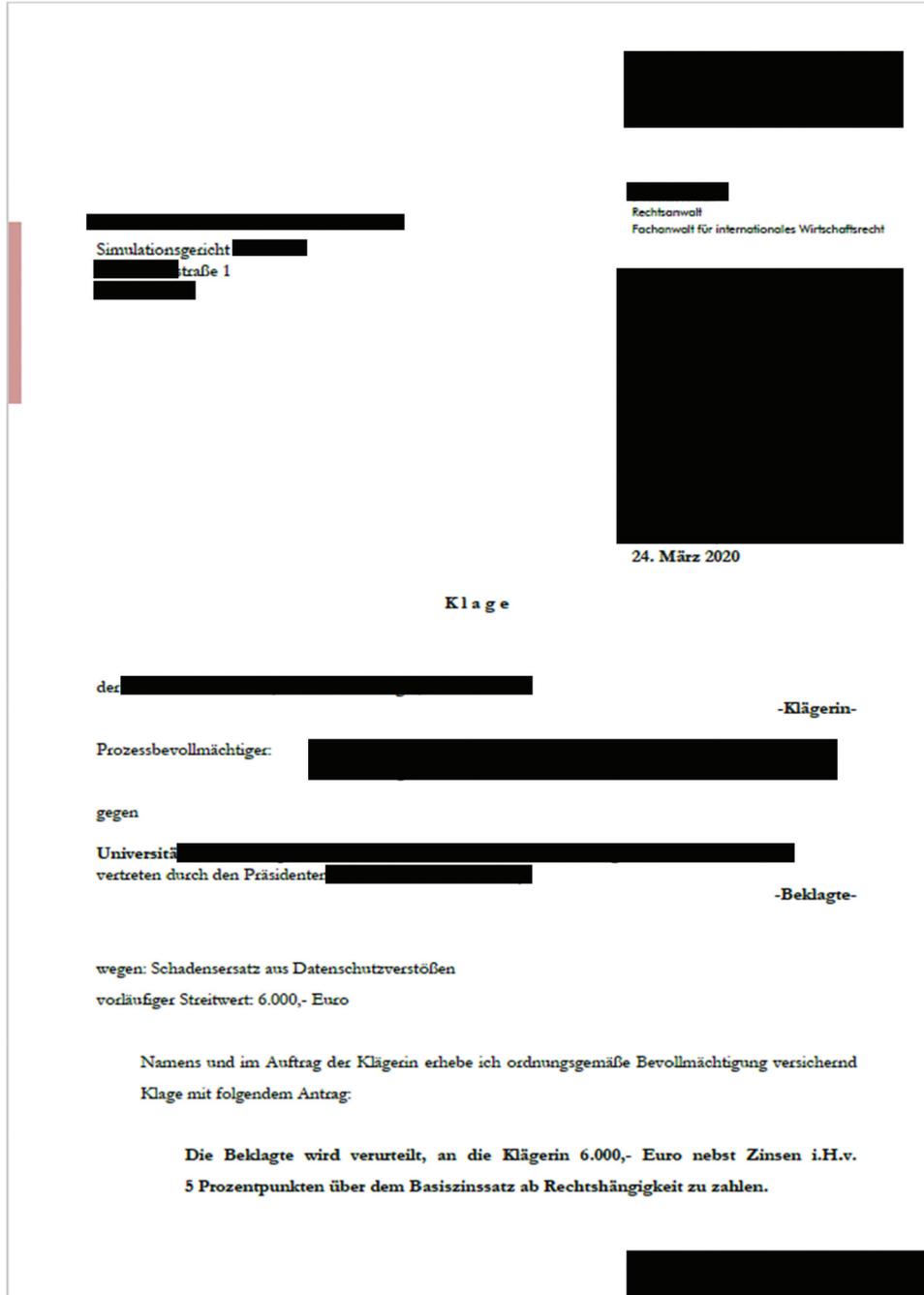


Figure C1. Statement of Claim.

[REDACTED]

Simulations-Verwaltungsgericht [REDACTED]
[REDACTED]

[REDACTED]

Klageerwiderung [REDACTED] 28.03.2020

In der Sache [REDACTED]

melde ich mich für die Beklagte, die [REDACTED] als
Prozessbevollmächtigter.

Die Beklagte lässt sich sehenden Auges auf die Klage vor dem Verwaltungsgericht ein.
Ordnungsgemäße Bevollmächtigung wird versichert und ist gerichtsbekannt.
Einer Übertragung auf die Einzelrichterin stimme ich Namens der Beklagten zu.
Einer Verhandlung durch Bild- und Tonübertragung stimme ich Namens der Beklagten zu.

[REDACTED]

Namens der Beklagten beantrage ich

1. die Klage abzuweisen;
2. der Klägerin die Kosten des Verfahrens aufzuerlegen.

Begründung

1. Zum Sachverhalt:

Die Klägerin studiert im dritten (3) Semester Wirtschaftsinformatik an der [REDACTED]
und hat im Wintersemester 2019/2020 die Veranstaltung [REDACTED] be-
legte. Zutreffend ist, dass zur Klausurvorbereitung vom 21. Februar 2020 bis zum 26. Februar
2020 vom Lehrstuhl [REDACTED] die Lernapplikation [REDACTED] eingesetzt wurde und die Klä-
gerin am 24. und 25. Februar 2020 an dieser Klausurvorbereitung teilnahm.

Seite 1 von 5

Figure C2. Statement of Defence.

[REDACTED]

[REDACTED]

[REDACTED]

Simulations-Verwaltungsgericht [REDACTED]

[REDACTED]

[REDACTED]

In dem Verwaltungsrechtsstreit

[REDACTED]

gegen

[REDACTED]

replizieren wir in gebotener Kürze. Wir behalten uns jedoch weitere Sachverhaltsdarstellungen für die mündliche Verhandlung vor, da nicht alle Aspekte in der vom Gericht knapp bemessenen Frist geklärt werden konnten.

I. Zum Auskunftsanspruch

(1) Es trifft zu, dass die Beklagte nun der Klägerin eine Auskunft über die gespeicherten Daten erteilt hat. Allerdings ist die von der Beklagten erteilte Auskunft ungenügend und widersprüchlich. Die Klägerin kann nicht überprüfen, ob die erteilten Auskünfte auch zutreffen und muss dies mit Nichtwissen bestreiten.

(2) Es bestehen aber Anhaltspunkte dafür, dass die Auskunft nicht vollständig ist. So führt die

[REDACTED]

Figure C3. Replication (reply of the plaintiff in civil proceedings to the defendant's statement of defence).