

Using Design Thinking for Requirements Engineering in the Context of Digitalization and Digital Transformation: A Motivation and an Experience Report



Angela Carell, Kim Lauenroth, and Dirk Platz

1 Introduction and Motivation

Digitalization and digital transformation are omnipresent terms inside and outside the software engineering community. Many governments consider digitalization of industry and society as the primary challenge of this decade (cf. [1]). However, people inside the software engineering community often consider both terms as buzzwords that do not provide meaning. They often argue that everything in software engineering is about digitalization and that digitalization and digital transformation are nothing novel for them. Unfortunately, this narrow perspective prevents the community from recognizing that the world outside the software community has a different understanding. This understanding reveals a significant change of the software business that has a tremendous impact on the way a software is developed.

This chapter, which draws upon practical experience, first shows in Sect. 2 how digitization, digitalization, and digital transformation have affected or are currently impacting the discipline of requirements engineering. In particular, the digital transformation that has begun is leading to disruptive changes here, as is also the case with many other areas of software engineering. A key finding is that requirements engineers have to adopt the mindset of designers to cope with the challenges that emerge from projects in the context of digital transformation.

In Sect. 3, we explain the methodology of design thinking as a current framework that has proven itself in practice to live up to the new required mindset. Then, we will present two concrete adesso AG project examples of how design thinking came to be employed at the company and the results it achieved in specific project

A. Carell (✉) · K. Lauenroth · D. Platz
adesso AG, Dortmund, Germany
e-mail: angela.carell@adesso.de; kim.lauenroth@adesso.de; platz@adesso.de

situations. These results were quite surprising from the point of view of a “classical” requirements engineer. Section 4 contains the summary and the conclusions.

2 From Digitization to Digital Transformation

The way software and digital technology impacts business and society can be separated by three different terms: digitization, digitalization, and digital transformation. Inspired by the work from [2], we use these three terms to form three levels of impact of digital technology. These three levels can be considered at the same time as a kind of historical development. However, these three levels are a mental model that is used to characterize and explain phenomena that the authors have observed. They should not be understood as a strict framework that allows for a precise classification. Instead, these levels should be considered as a way of looking at the impact of digital technology that helps to understand the changes that the authors recognized in their professional life.

2.1 Level 1: Digitization

Digitization literally means the conversion of analogue information into a digital (binary) representation [3]. This conversion is a prerequisite for making information processable for digital technology. Processing large amounts of information was one of the primary reasons for developing computers [4]. The impact on business and society on this level is limited; the digital technology is mainly used inside organizations without much visual surface to end customers and society.

Examples of digitization are:

- Banking: Bills are no longer paid with cash, but with credit cards or electronic cash cards.
- Business Administration Systems: Several industries use computers to manage their business, for example, insurance companies manage contracts and claims with software.
- Office software: Letters and documents are written by using a dedicated office software and are no longer written on paper or with a typewriter.
- Records to compact discs (CD): Music is no longer stored in an analogue way on records. The music industry now sells compact discs with digital music.
- Analogue to digital photography: Digital cameras provide a new way of taking pictures.

The essential characteristic of digitization is that there is an analogue model of the process or artifacts that is digitized. The user of this new digitized product or service typically recognizes only a minor change. For example, the early digital cameras had the same format as the analogue cameras. Applications for insurances

or insurance claims were made with paper-based forms that were digitized later in the insurance company by an insurance clerk.

From a software engineering perspective, the challenge was to *understand the analogue product or service on a detailed level to create the digital equivalent*. The challenge has led to the development of the software engineering discipline requirements engineering (RE). The focus of RE is the proper understanding of stakeholder (i.e., customers, users, etc.) requirements and the documentation of these requirements to make them available for a structured software engineering process [5].

2.2 Level 2: Digitalization

With the increasing power of computer hardware and the advent of broadband Internet connection at the end of the 1990s, a new level in the use of software and digital technology can be recognized. The term digitalization is often used to describe this phenomenon (cf. [6]). Instead of focusing on the transformation of information into a digital format, whole processes and businesses start to move from the analogue world to the digital world. The impact on business and society can be considered medium, underlying business models and the society remains unchanged, the main goal is to improve the existing products and services. The result of this development is that people were increasingly exposed to digital technology in several places of their daily and professional lives in the late 1990s. Examples of digitalization are:

- Online shopping: customers buy various products over the Internet.
- Online banking: customers start doing their bank business over the Internet.
- Online music: customers buy music over the Internet and listen to music with a digital device (e.g., MP3 player).

This transition from digitization to digitalization appears to be minor from a technical perspective, since the devices remain unchanged. They became more powerful in terms of processing power, memory, storage, and network capabilities. From a software engineering perspective, the challenge of understanding the business remains, but two new challenges emerged in this phase.

Firstly, the software in the context of digitalization is mainly used by people that were unexperienced and often novices in terms of computer and software. Therefore, the software had to be designed in a way that enables intuitive usage and supports the user as much as possible. This challenge eventually led to the establishment of usability engineering in software development [7].

Secondly, potential products and services that were suitable for digitalization must be identified prior to the development. This task constituted a significant challenge because *knowledge about the business must be combined with a proper understanding of the potential benefits of digital technology*. History showed that this task was very difficult and that the potential of digital technology was

overestimated by orders of magnitude in the late 1990s. The peak of this negative development was the so-called dot-com bubble [8].

2.3 Level 3: Digital Transformation

The growing dissemination of digital technology (e.g., smartphones, mobile Internet, portable computers) in the late 2000s and the successful digitalization finally led to a phenomenon called digital transformation. Digital transformation is characterized by significant changes in business models and in society enabled by digital technology (cf. [9]). Examples of digital transformation are:

- Crowdfunding: A project, product, or venture is funded by many people. Software platforms (e.g., Kickstarter or Indiegogo) made this concept popular and provide a service for presenting ideas to people and allowing them to support the idea with a certain amount of money. Crowdfunding provides a real alternative business model for project funding compared to credit-based funding from a bank (cf. [10]).
- Music streaming flat rate: The customer pays a fee (typically monthly) to a platform provider (e.g., Spotify) that allows the customer to listen to all the music provided by the platform. The central changes in the business model are that the customer no longer buys a certain song but pays a fixed fee and that artist receives payment based on the number of listened songs.

From a technical point of view, the difference between digitization and digitalization again is minor since the devices remain more or less unchanged besides further increases in power (e.g., for streaming large amounts of music data).

On the level of digitalization, existing business models or products were improved, that is, it was clear that the underlying business is sustainable (e.g., selling books). The major challenge was to identify a proper way for the digitalization of the business. On the level of digital transformation, this underlying assumption is no longer valid and an additional challenge emerges: *changes in business models and society are only successful if people (customers/users) see value in a new digital product or service* (e.g., paying money every month for listening to music). The real user needs play an essential role. Users had to adopt their way of working to the systems capabilities (good requirements engineering was useful to bring the system as close as possible to the user's requirement). Nowadays, fundamental user needs (that are not addressable by asking) must be identified to be successful.

2.4 Conclusion: The Growing Need for a Holistic Design Competence in Software Engineering

From a technical perspective, the three levels of digitization, digitalization, and digital transformation are equal or at least very similar. Devices such as computers,

smartphones, or tablets are used to power software that has been developed to perform one or more tasks in a certain context. The technical development of software (i.e., the act of coding and selecting technical realization alternatives) remains more or less unchanged since the medium software itself does not change. From a historical perspective, the major change is the growing power of the devices, the increased mobility and connectivity of new devices, and the development of new programming languages (e.g., Java in the late 1990s) (cf., e.g., [4]).

From the outside perspective, the three levels show a significant difference in their impact on the context in which the software is developed:

- Digitization means that an analogue information medium changes to a digital medium within an otherwise stable context (e.g., maintaining insurance policy in a software database instead of a paper folder).
- Digitalization means that analogue processes are replaced or extended with digital processes within an otherwise stable context (e.g., doing bank business via the Internet instead of going to a bank's office).
- Digital transformation means that the whole context is changed by digital technology (e.g., funding a project via Kickstarter instead of applying for a credit).

The main conclusion from this list is that each level increases the scope of what is covered by the development of the software. In digitization, the software development can fully rely on the context and can focus on the proper software representation of the analogue model. In digitalization, software development must come up with a proper idea of the digital process for an existing analogue process. Here, software development has to become creative together with the business to develop an optimal solution. Typically, there is an analogue model of the process, but a simple transfer of the analogue process is typically not feasible, since digital technology offers different possibilities (e.g., buying books over the Internet is a different experience compared to a physical bookstore). Finally, digital transformation has the largest scope since the business model and the digital product/service determine each other and are created in parallel. In this situation, there is no analogue model that can be used as a reference point for the development activities. Instead, the business model and the software must be developed in parallel since both influence each other.

Development paradigms that fully rely on business stakeholders (or product owners) to provide the requirements for the software under development are not appropriate for digitalization and digital transformation since stakeholders typically do not have a profound understanding of the technical capabilities of software. On the other hand, trained software developers typically do not have a proper understanding of the business context to come up with optimal ideas. Hence, these development paradigms suffer from a competence gap since business people typically only focus on the business side and software people typically only focus on the technical side of the software.

This chapter argues that software development must drop the assumption that there is somebody out there that has a proper understanding of what shall be

developed. Instead, software development should start to guide the process of designing a software in the sense of industrial or product designers. Industrial designers have a holistic view on a product and feel responsible for the overall shape of a product (appearance, functionality, and quality aspects). They interact with stakeholders and create ideas for the product instead of purely collecting requirements from the stakeholders (cf. [11]).

This paradigm shift has a significant impact on software engineering and especially on the disciplines of requirements engineering and usability engineering, since these disciplines are the interfaces to the stakeholders and the front-end of the software development process. The central change is the way requirements and usability engineers have to work and interact with their stakeholders. Instead of focusing on the collection, documentation, and validation of requirements, both disciplines have to participate actively in the creation process of the software.

3 Design Thinking as a Method to Think About Software

Implementing the paradigm shift is by no means an easy task. It requires people that use designer's ways of working (cf. [12] for a comparison of designer's working style with other disciplines) and substantial methodical support. Design thinking is a framework that has the potential to provide this support in terms of principles, process models, and techniques. The following subsections will provide a brief overview of design thinking and present two case studies for the successful application of design thinking in two development projects of the adesso AG.

Besides design thinking, there are other methods with similar goals, for example, contextual design [13], design sprint [14], or liberating structures [15]. A comparison of such methods is not the objective of this chapter. An overview of creativity/design techniques in requirements engineering can be found, for example, in [16].

3.1 A Brief Overview of Design Thinking

Design thinking can be defined as a methodological approach (a framework). Above all, though, it is an attitude amenable to consistently developing innovation potential and new solutions from the users' perspective.

Design thinking is a human-centered approach to innovation that draws from the designer's toolkit to integrate the needs of people, the possibilities of technology, and the requirements for business success. (Tim Brown, president and CEO of IDEO) [17]

This approach has been in development since around 1996 and emerged from the collaboration of Terry Winograd and Larry Leifer with David Kelley (Stanford

University) and the design company IDEO. In 2005, d.school was founded at Stanford University, which helped to spread the idea of design thinking.

Design thinking is based on specific principles that differentiate the process from other creative processes and make it particularly useful for an IT context. These are described below.

Principle 1: Users' Needs Take Center Stage

Design thinking involves a human-centered design approach. This means that the needs of the users are placed at the center of attention and the design is consistently viewed from the perspective of future users. This approach is particularly relevant for projects in the context of digital transformation, since the success of such projects is decisively dependent on the acceptance of the end user.

Empathetically “stepping into users’ shoes” and understanding their needs and wishes are essential aspects of this approach. However, design thinking goes far beyond classic human-centered design approaches: It delves much deeper into the needs of the users and attempts to uncover their latent needs, that is, those that cannot be articulated. The findings thus obtained provide important information that can be used to develop creative solutions and discover innovative potential.

Deep empathy for people makes our observations powerful sources of inspiration. We aim to understand why people do what they currently do, with the goal of understanding what they might do in the future. [18]

Unlike participatory design [19], in which the users actively participate in the development of the solution, the actual generation of ideas takes place without direct user involvement. Representatives of design thinking assume that users themselves often have no access to their real needs. Asking them directly about ideas will hardly yield any revolutionary, new solutions. For example, the following quote has been ascribed to Henry Ford: “If I had asked people what they wanted, they would have said faster horses.” At the same time, though, design thinking does not mean that the user is kept out of ideas development. Instead, the developed solutions are evaluated step-by-step with users and thus systematically refined and further improved upon. Users are not asked directly for solutions, though.

Principle 2: Deep Understanding Rather Than Large Numbers of Cases

In contrast to other methods, design thinking does not rely on large-scale qualitative user surveys or questioning focus groups. These have little use in design thinking, as this approach is not about identifying aspects that are equally common to *most users*. Instead, design thinking aims to discover interesting, surprising, astonishing, or even irritating things and to take these as starting points for developing innovative ideas. The process is similar to a gold digger digging for small gold lumps (nuggets); design thinking involves searching for nuggets that can act as starting points for creative solutions. This is why the term “nuggeting” is used in this context.

In order to find these “innovation nuggets,” persons from extreme groups are often surveyed or observed. Extreme groups consist of persons who either do something extremely often or intensively or take a position of extreme denial. The idea behind observing these groups rests on the assumption that these persons

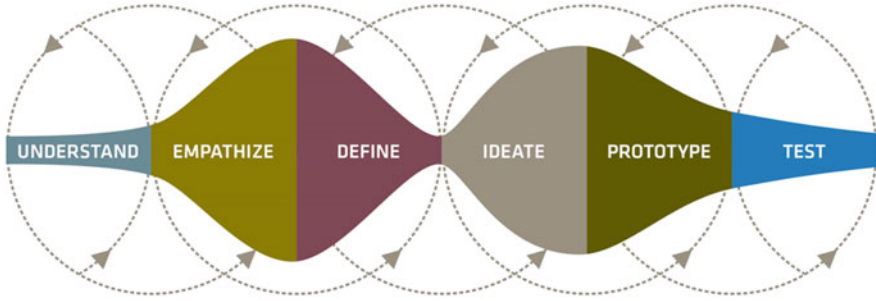


Fig. 1 The design thinking process

exhibit behavior and reveal needs that are also present in other user groups, but in a much weaker form, and are therefore very difficult to access.

Principle 3: Interdisciplinary Team

The team represents a central element of a successful design thinking project. It should consist of members who are open to new things and dare to try something new. They must be able to look beyond their own ranks, to appreciate the opinions and perspectives of others, and thus to constructively deal with them. Ideally, teams are recruited from different areas to create a diversity of perspectives—this is the only possible way to tackle problem-solving in a manner that is “different from the norm.”

Principle 4: Follow a Clear Process

Design thinking follows a process [20] that consists of both divergent and convergent phases (Fig. 1). Divergence is based on diversity and illuminating a problem from different perspectives. Convergence leads to a consolidation and merging of ideas all the way up to the solution. The process is not to be understood as a rigid linear sequence; it may include feedback loops and iterations.

Phase 1: Understand

In this phase, the main focus is on understanding the problem that is posed (this is also called a challenge) and its essential elements. In design thinking, it is assumed that the problem formulation at the beginning of the design thinking process can only provide a rough description of the problem area. It can only be refined and fleshed out in the course of the process on the basis of the insights gained. In this phase, however, an initial understanding of the problem context is wholly sufficient. In terms of the methodology, the idea is to mainly research data and facts, identify relevant actors and situations, and explore the possible scope of design.

Phase 2: Empathize

This phase is essentially about building up empathy with users in order to understand their views of the world or the problem. This phase expands the focus (divergence) set by the challenge and provides deep insights into what users think and feel and, above all, where their real needs lie. Tim Brown describes this very aptly: “We need

to return human beings to the center of the story. We need to learn to put people first” [21]. This involves techniques that are particularly suited to building empathy, such as the empathy-oriented interview, observation, and active entry into a situation to experience it through the user’s eyes [22].

Phase 3: Define

At this point in the design thinking process, all the information and insights gathered so far are examined in detail, patterns are identified, and, above all, unexpected and surprising aspects are worked out (nuggeting). In contrast to a cross-sectional analysis of interviews, during this phase, all team members share all their impressions and information with each other. The collected impressions and experiences are used to reformulate and specify the initial problem (convergence). This refocusing may even be quite radical, namely, when the presumed cause of a problem proves to be incorrect.

‘If a problem is not worth solving, it’s not worth solving well.’ Focusing our energy on the right question can make the difference between incremental improvement and breakthrough innovation. [18]

Phase 4: Ideate

The idea-finding phase begins with the reformulated challenge (point of view). First of all, personae (prototypical users) are developed in order to be able to grasp the target group for the future solution in as concrete of terms as possible. Personae are used to repeatedly compare the solutions to be developed against users’ needs. In the next step, as many different ideas as possible (divergence) are developed, based on the focused challenge and the defined persona(e). For example, creative techniques can be used, ranging from purely intuitive (e.g., brainstorming, analogy, bisociation) and discursive techniques (e.g., Osborne checklist, Ishikawa) to combinations of both (e.g., Walt Disney method, Triz) [23]. The objective of using these methods is to resolve so-called priming effects—priming refers to the unconscious activation of certain associations while generating ideas due to previous experience or other impulses. After a large number of ideas have been collected in this way, the solutions are evaluated, and the promising ones are selected.

Phase 5: Prototyping

Once ideas are identified that are to be pursued further, the process moves on to implementing concept prototypes. Unlike in software development, this is generally not a viable piece of software or any other concrete modeling of an IT system (e.g., [mock-up](#)). It is rather an initial design thinking prototype that serves to visualize an idea quickly and concisely, in such a way that it can also be rejected quickly. A prototype can therefore be made from a variety of materials (e.g., paper and cardboard or Legos). It can also be a role play, a storyboard, and so on. It is essential that these concept prototypes offer the user possibilities for interaction.

Phase 6: Test (Trial)

Advanced prototypes are tested in a real context during the test phase. Unlike prototyping, it is crucial here to test the prototype in the context in which it is to be actually used later on. The context or the testing in the real situation once again

creates empathy for the target group and their needs and provides insight into the context-related factors to be considered for the final solution.

In the following section, two example projects will show how design thinking is used in the practice of an IT company.

3.2 Example 1: Online Jewelry Shopping

The Challenge

A jewelry chain (the customer) has been selling its goods (watches, jewelry) through its conventional chain store. For some time now, it has also been selling them via an online marketplace as well as a mobile channel. The sales figures for the mobile channel in particular have lagged behind the customer's expectations. To address this issue, initial approaches and ideas for optimizing the online business were to be worked out within the framework of a design thinking project. The challenge was thus as follows: "How can we make more people buy jewelry through the mobile channel?"

Design Thinking Setting

The design thinking project was tailored to fit into 5 workshop days. The design thinking team consisted of 12 people from various different professions (three customer representatives from the area of customer IT, two Web designers, two app developers, two concept developers, one secretary, and two moderators).

The Design Thinking Process

After a concise customer briefing, the team worked on the topic of "jewelry retail" and became familiar with products, services, target groups, and competing portals ("understand" phase). For the observation phase ("empathize" phase), it was decided to first determine customers' needs related to the jewelry purchase independently of the actual sales channel, as well as to gather information about what people associated with jewelry—including what positive and negative experiences they had with jewelry and jewelry purchases. The way in which the interviewees purchased jewelry was irrelevant for this approach at that point, and so the interviews were conducted on a Saturday morning on a shopping street of a major German city. Passers-by were interviewed at random. Back at the workshop, the collected findings from the stories interviewees told were shared within the team and parsed for contradictions, surprising insights, and amazing (nuggeting) factors ("define" phase). Surprisingly, many people wanted to express their particular appreciation for the person to be presented with the jewelry through their choice and purchase of a piece of jewelry. The original challenge was revised on the basis of this insight. It was no longer generally a matter of getting more people to buy via the mobile channel. Instead, it was much more precisely about the question of how to create value for the recipient through the purchase of a piece of jewelry. This new challenge prompted the team to think about online portals that are designed more like treasure

or memorabilia boxes, or where jewelry can be linked with individually stored images of other memorabilia.

Conclusion

The design thinking project yielded more than 250 ideas for a future mobile shop for the customer. Not a single line of code was written, nor were wireframe or mock-up models created. The IT experts involved had to fully accept this process that had nothing to do with IT. However, their IT know-how was still very valuable to the process, and they themselves learned a lot about the potential customers of the future IT solution—more than they would have ever learned in any other project. They continued to maintain this customer-centric perspective during the subsequent software development process.

3.3 Example 2: Developing Innovative Software for Dentists

The Challenge

For this design thinking project, the focus was on developing a new and, above all, innovative software for dentists. The existing IT solution was already highly outdated, its user interface was very inconvenient, and, above all, the software failed to sufficiently meet the legal requirements for documentation and quality assurance. Within the framework of the design thinking process, a viable innovative idea was to be developed within 3 months from scratch, then tested on initial IT prototypes, and worked out in the form of a rough specification. The challenge was as follows: “How can we create an IT system for dentists that significantly reduces the cost of documentation for them?”

Design Thinking Setting

For this project, the actual design thinking team consisted of IT experts, educators, dentists, and social scientists. The customers to be surveyed were specifically selected by the client. The prerequisite was that both dentists themselves as well as nonmedical specialists (users of the future solution) were to be surveyed. In addition, both groups were to be as heterogeneous as possible in terms of age, gender, and size of practice, with attitudes ranging from “highly professional” to “pragmatically relaxed.” Three-day workshops were held independently with each group.

The Design Thinking Process

First, persons affected by (stakeholders) and users of a new dentistry system were identified in working with both groups (“understand” phase). The design team was surprised to learn that apart from the obvious groups (such as dentists and dental assistants), cleaning specialists had been named as well. After all, the cleaning process and the cleaning agent used must also be documented via the software, according to the reasoning. In the next step, personae were developed for the main stakeholders. These were to be designed in such a way that they would differ sig-

nificantly in their professional attitudes, views, and working methods (“empathize” phase). The workshop participants were able to draw on the abundance of their daily experiences and worked very intensively on creating the most detailed and accurate characterizations possible. Subsequently, the participants developed a key set for each persona, which summarized the general requirements for the software from the perspective of said persona (point of view: How can we make it happen that persona X . . .). These statements were elaborated amidst intensive discussions, which generated many insights for the design thinking team (“empathize” phase). Finally, a series of ideas were generated for each persona during the ideation phase. Prototypes of these ideas were implemented selectively with paper and cardboard (“prototyping” phase). The group discussed the solutions with verve and commitment. The design thinking team questioned the solutions over and over again and wanted to know, above all, *why* a solution was supposedly suitable for a certain persona. At some point in the process, one participant erupted in frustration:

He [dentist persona] only wants to work with his patients—he doesn’t want to document anything at all. He became a dentist because he enjoyed working with patients. He doesn’t want to be an expert in dentistry IT.

The design thinking team then developed an idea of how the IT system could be used to inform patients about their treatment as best as possible on the basis of pictures and other visualizations. Documentation, then, merely occurred in the background.

Conclusion

In this process, the software developers worked intensively for a very short time (two periods of 3 days each) on the conditions in different dental practices, taking legal requirements and practical concerns of the day-to-day business in these practices into account. The workshops with the dentists and assistants were not seen as a participatory development process, but rather as a source of inspiration for developing a truly innovative solution for the dentistry software of the future. The empathy developed during the design thinking process for the users and their needs was felt all the way through to the creation of the specification document: It was examined over and over again and questioned as to whether it was still on target where needs were concerned.

4 Summary and Conclusions

We have shown that the different levels of digitalization require correspondingly adapted procedures or methodologies for requirements engineering. This requires new frameworks or models for projects in the context of digital transformation. This is particularly due to the fact that these projects call for specifying completely new business models and realizing these by means of IT, which can only succeed if users’ actual needs are met and satisfied as best as possible. One of the main challenges is

that these needs often cannot be identified by asking questions, but must instead be tracked down.

Design thinking presented in this chapter serves as a suitable framework for this purpose; adesso has successfully used this method in many project situations within the context of digitalization and digital transformation. We have illustrated this by providing examples of two real projects. One of the main advantages of this methodology is that people and their needs are at the center of the design process.

In practice, we have found that it is helpful in such projects if, in addition to the classic management roles (a project manager responsible for budget, quality, and time and an architect responsible for the technical implementation), a third management role is established with responsibility for designing software according to the subsequent user's requirements.

The third management role is key to:

- Ensuring the necessary focus in these projects on the design of the software from the point of view of the user
- Avoiding conflicts of interest as far as possible, because an architect may, in case of doubt, decide against the user of the software and in favor of a simple technical solution

These three management roles should already be filled at the start of the project and cooperate with each other in the course of the project according to their responsibilities.

As with all frameworks and methods, when each new project is initialized, one must examine the extent to which design thinking is adapted to the project context and to what extent project-specific customizing of the methodology is required.

We are doubtless still at the beginning of the digital transformation and thus also at the beginning of the further necessary changes to the mindset of successful requirements engineers. The particular goals are to create new software systems that are very popular among users and, consequently, to successfully establish new business models in the marketplace from the point of view of the respective client.

References

1. Digital Agenda, Federal Ministry for Economic Affairs and Energy. <https://www.bmwi.de/Redaktion/EN/Artikel/Digital-World/digital-agenda.html>, last accessed 2017/07/19
2. Khan, S.: Leadership in the digital age – A study on the effects of digitalization on top management leadership. Master Thesis, Stockholm Business School (2016)
3. Merriam-Webster Dictionary. <https://www.merriam-webster.com/dictionary/digitization>, last accessed 2017/07/19
4. History of Computer. <http://www.computerhistory.org/timeline/computers/>, last accessed 2017/07/19
5. Pohl, K., Chris, R.: Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam – Foundation Level – IREB Compliant. Rocky Nook (2015)

6. Vogelsang, M.: *Digitalization in Open Economies, Contributions to Economics*. Physica-Verlag, Heidelberg (2010)
7. Nielsen, J.: *Usability Engineering*. Morgan Kaufmann (1993)
8. DotCom Bubble, Wikipedia. https://en.wikipedia.org/wiki/Dot-com_bubble, last accessed 2017/07/19
9. Kane, G., Palmer, D., Phillips, A.N., Kiron, D., Buckley, N.: *Strategy, Not Technology, Drives Digital Transformation*. MIT Sloan Management Review (2015)
10. Ordanini, A., Miceli, L., Pizzetti, M., Parasuraman, A.: *Crowdfunding: transforming customers into investors through innovative service platforms*. *J. Serv. Manag.* **22**(4), (2011)
11. What is ID? Industrial Designers Society of America. <http://www.idsa.org/events/what-id>, last accessed 2017/07/19
12. Durling, D., Cross, N., Johnson, J.: *Personality and learning preferences of students in design and design-related disciplines*. IDATER 96 (1996)
13. Beyer, H., Holtzblatt, K.: *Contextual Design. Defining Customer-Centered Systems*. Morgan Kaufmann (1997)
14. Knapp, J., Zeratsky, J., Kowitz, B.: *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*. Simon & Schuster (2016)
15. Lipmanowicz, H., McCandless, K.: *The Surprising Power of Liberating Structures: Simple Rules to Unleash A Culture of Innovation*. Liberating Structures Press (2014)
16. Lemos, J., Alves, C., Duboc, L., Rodrigues, G.N.: *A systematic mapping study on creativity in requirements engineering*. In: *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pp. 1083–1088. ACM (2012)
17. Homepage. http://pure.au.dk/portal/files/45289412/Bilag_11_Om_IDEO.docx, last accessed 2017/07/20
18. Kelley, T., Kelley, D.: *Creative Confidence. Unleashing the Creative Potential Within Us All*. HarperCollinsPublishers, London (2013)
19. Kensing, F., Blomberg, J.: *Participatory design: issues and concerns*. *Comput. Supported Coop. Work.* **7**, 167–185 (1998)
20. Kelley, T. (with Jonathan Littman): *The Art of Innovation*. HarperCollinsBusiness, London (2001)
21. Brown, T.: *Change by Design*, p. 39. HarperCollinsPublisher, New York (2009)
22. Hasso Plattner Institute of Design at Stanford. <https://static1.squarespace.com/static/57c6b79629687fde090a0fdd/t/58890239db29d6cc6c3338f7/1485374014340/METHODCARDS-v3-slim.pdf>, last accessed 2017/07/31
23. Kumar, V.: *101 Design Thinking Methods. A Structured Approach for Driving Innovation in Your Organization*. Wiley (2012)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

