Please quote as: Li, M. M. & Peters, C. (2016): Mastering Shakedown Through The User: The Need for User-Generated Services In Techno Change. In: European Conference on Information Systems (ECIS), Istanbul, Turkey.

MASTERING SHAKEDOWN THROUGH THE USER – THE NEED FOR USER-GENERATED SERVICES IN TECHNO CHANGE

Research in Progress

Li, Mahei Manhai, Research Center for IS Design (ITeG), University of Kassel, Germany, mahei.li@uni-kassel.de

Peters, Christoph, University of St. Gallen, Switzerland, christoph.peters@unisg.ch & Research Center for IS Design (ITeG), University of Kassel, Germany, christoph.peters@uni-kassel.de

Abstract

Turning IT -investments into organizational benefits is one of the key research objectives for information systems. There is a significant body of knowledge dealing with research on system development and management approaches. However, by leveraging crowd mechanisms, we believe that from a service systems perspective, after the rollout (also called shakedown phase) organizations still require active involvement by end-users in order to create IT-impacts and ultimately enhance organizational performance. We follow a design science research approach and suggest integrating user-generated services into an existing service system. Our artifact will hence consist of design requirements and design principles, which we will validate in our future work through two hermeneutic circles, in which users are given the means to identify existing problems and create their own fitting solutions. The focus of this research-in-progress paper lies on identifying design requirements that address the challenge of motivating users to create and use user-generated services. Our first step of developing design requirements and preliminary results are based on a brief literature review to identify existing motivational factors and incentives from the software implementation literature, which is the basis for our design requirement for user-generated services. Our theoretical contribution lies both in showing the potential of user-generated services for the shake-down phase for techno change projects and providing a first tentative motivational design requirement for a service system that enables user-generated service in the context of the shakedown phase.

Keywords: user-generated services; service systems engineering; shakedown phase; design science research

1 Introduction

Even though the benefits of information technology (IT) projects can be alluring, the experience is all too often disappointing, since the implementation of new technology often poses serious challenges to companies. Organizational change projects involving the introduction of a new technology have up to 75 percent chance of failure, a problem that has been around for over a decade (Majchrzak, Meshkati 2001).

In regards to the question how to create business value, there are three main processes that we adapt: the IT-conversion, the IT-use, and the competitive process (Soh, Markus 1995). Value propositions reflect the potential of any IT project and one taps into the dormant potential if someone uses the new system as was originally intended (Chandler, Lusch 2015). Our overarching research objective is therefore to increase the intended use of software in order to realize the initially defined value proposition, which is focus of the IT-Use process (Markus et al. 2000). We focus on a specific type of IT projects: techno change projects, which are IT-projects that have a high organizational impact and combine both traditional IT-management with change management challenges (Markus, 2004). Here, negative attitudes of people towards change can disrupt the assimilation of new technological resources into their day-to-day operations, which we describe as the *shakedown* phase. We identify the shakedown phase as a key challenge that we address from a service system perspective. To clarify, during this phase, the rollout already occurred and the organization is facing following daunting task: it needs to start using the new software solution for their daily business. Since the software is new, the organization only has few business-specific documentations, incident reports and experts. The few experts that do exist most likely acquired their knowledge through early involvement during requirements elicitation and implementation of early pilots. Neither their number, nor their knowledge and training are sufficient to support the vast information needs of their broad user base (Markus 2004). Furthermore, without existing support, software implementation is unlikely to fulfill its potential. It is therefore not only a matter of identifying their acceptance, but rather how to make new users quickly learn to use the software appropriately as initially intended (Soh & Markus 1995). Successful implementation of software into an organizational context is hence heavily reliant on dedicating resources specifically for transferring and integrating knowledge among its actors (Roberts et al. 2003). In addition, incremental and ongoing improvements need to be made to both culture and technology. An emergence perspective has to be implemented in addition to the planned approach of technological or cultural/change management-driven approaches (Jackson, Philip 2010). To address said challenges, we believe that active user involvement during the shakedown phase is decisive.

We apply a service systems perspective, because of its inherent focus on finding the right configurations of both IT and non-IT resources (actors) in order to create value in the right context through the use of services (Lusch et al. 2008; Maglio, Spohrer 2008; Vargo, Lusch 2004; Böhmann et al. 2014b). Recent research revisits the importance of value propositions and the engagement of service systems (Chandler & Lusch 2015), especially for the role of corresponding business models (Peters et al. 2015). The right configuration of services with actors ("who") within a service system fits our approach of engaging and connecting users ("who" and "with whom") through user generated services (artifact) during the shakedown phase ("when") (Chandler & Lusch 2015, p.1).

There has been abundant research that demonstrates the importance of integrating end users. Most focus on the software development phases (Adolph et al. 2012) include agile methods (Highsmith, Cockburn 2001) and are even holistically value-driven (Mueller, Thoring 2012). All have one underlying assumption in common: involving users earlier and more often. Yet, there has not been much research on designing actual means to increase the success of the shakedown phase through higher user involvement. Exceptions are some knowledge management tools, such as wikis that solve ad-hoc and distributed needs of users (Wagner 2004). The transfer of knowledge is an integral part of any shakedown phase. Knowledge management approaches and tools are often used for enterprise systems and their implementation (Vandaie 2008). However, from a service systems perspective, we propose the

notion of user-generated services to involve end-users directly in the software implementation process. We assume that through the wisdom of the crowd in organizations (Ebner et al. 2008; Surowiecki 2006)the potential of users knowing how to solve certain context-specific problems can be leveraged for the organization. By providing the necessary tools to the users, they can create their own services and provide them to their peers. This empowers and actively integrates the users into the software implementation process. In our research context, software implementation is unlikely to fulfill its potential without proper user support. Techno change projects often fail to realize the initially promised benefits, resulting in the benefit capture phase being no more than fiction. Service support is furthermore an important aspect of creating routine, so that the newly implemented service system is not only accepted but also used as intended (Zmud & Cox 1979). We propose motivational user-generated services (UGS) to address the problem of insufficient training and documentation (Markus 2004).

We follow design science research principles (Hevner et al. 2004). The tools for UGS and the integration method into an existing service system are therefore our resulting design artifacts. However, since we are researching the means of creating and integrating UGS, we must first understand how to motivate users to create and share services. In other words, to increase the likeliness of a successful user-generated service support, we must understand the motivational factors that govern the behavior of users during software implementation. As an initial research step and its main contribution, this paper examines possible motives and incentives that increase user involvement for software implementation projects. We conduct a brief literature review to gain a status-quo on existing motivation and incentive findings currently employed in software implementation literature.

The research goal of this paper is twofold: First, we present our overarching research design following a design science research (DSR) approach in order to design UGS for software implementation. Second, we want to identify motivational requirements for UGS that enable a more effective shakedown. As regards the second objective, the existing literature of software implementation has either focused on its organizational context (Silva & Hirschheim 2013) or on key success factors of implementation projects (Kim et al. 2005), e.g. how user involvement and trainings influence implementation success. However, we did not find any research that actively engages users during the implementation endeavor. This paper identifies motivational factors that are relevant for users to increase the success of the shakedown phase. In our future work, we will transform motivational factors and bundle them into activation supporting components, which will be prototyped and integrated into our overall research design, resulting in following research questions:

RQ1: How can we apply DSR for designing user-generated services for software implementation?

RQ2: What are motivational design requirements to increase interaction with user-generated services?

2 Theoretical Foundation

2.1 Service Systems Engineering

Current research already includes numerous approaches for the systematic development of single services (Bullinger, Scheer 2003), with relatively few addressing the context of a service system (Ernst et al. 2016). However, there are no methods that enable contextual co-creation of value with different stakeholders or interactions between service bundles (Böhmann et al. 2014). Different opinions of well-known researchers emerged, demanding that service research should assume a systems perspective (Maglio, Vargo 2009; Alter 2012). Service systems are systems that align service propositions with co-creation of value through specific configurations of actors and resources (Böhmann et al. 2014; Vargo, Lusch 2004) with actors being people that make use of their knowledge and skills and with the configuration of other actors and resources co-creating value (Maglio & Spohrer 2008; Alter 2013). Service systems describe these kinds of interactions, which in turn are embedded in value-generating processes. To be more precise, these service systems are socio-technical systems that ena-

ble co-creation of value (Böhmann et al. 2014). As mentioned in the introduction, Chandler and Lusch (2015) have stressed the importance of the right configuration of actors (with whom), while including the time component as context. Our research project is set at a certain phase of the lifecycle of a service system, the shakedown phase. We design and observe user-generated services and integrate them into an existing service system witch techno change characteristics.

2.2 Techno Change

There is much of research on software implementation approaches, either as a technical IT project perspective (Highsmith, Cockburn 2001; Mueller, Thoring 2012) or as an organizational change problem (Zmud, Cox 1979). Techno change is defined as "technology-driven organizational change" (Markus 2004, p.4). Its focus lies on organizational performance through a socio-technical and thus a service system (Böhmann et al. 2014). It integrates actors, organizational structures and IT-resources to create value and focuses on a holistic perspective concerning success factors. The key concepts of techno change is bridging the gap of only focusing on either the technological or the change management perspective (Markus, 2004). Organizations should therefore create an emergent environment that enables perpetual and iterative improvements to both the technology and organization (Jackson, Philip 2010).

Techno change distinguishes four main phases, each with its own goals, activities and problems: (1) chartering, (2) project, (3) shakedown and (4) benefit capture. During (1) the "idea is proposed, approved, and funded" (Markus 2004, p.11), next the actual technology is developed and shipped and in the shakedown phase, the organization starts using it, preferably reaching optimal performance as fast as possible. The optimal use and value creation then happens during the benefit capture phase. There are two main reasons as to why techno change is an appropriate framework while studying user-generated services: (1) Techno change projects are complex and need a holistic approach to deal with the challenge of shakedown (IT-use process)(Soh, Markus 1995). (2) We believe that the shakedown problems, as described by techno change, can be addressed by activating users in the existing organization.

Since techno change projects are inherently complex and aligned with organizational goals, the risk of failing the project is high. Techno change projects would greatly benefit from user-generated services during the shakedown phase. By tapping into the knowledge of the organization, they can share the often hidden experts and hence hidden knowledge with their peers. By sharing knowledge on how to solve current pain points with fellow users, users participate in finding a solution (Locke 1968) and are hence more motivated to use the system. This would probably replace a part of existing training units with more contextual and on-point services. Additionally, the risks of misusing IT in techno change projects is high, since the individual resistance to change is a known pain point (Markus 2004; Markus et al. 2000). This resistance oftentimes leads to workarounds, so the originally intended use of the technology is not achieved. Another takeaway from the techno change stream of research is the importance of the iterative emergence of new improvements and the negative impact on success by top-down disciplinary management styles (Jackson, Philip 2010). Since user-generated services constitute a bottom-up approach that is based on innovation by end-users, we firmly believe that techno change is an appropriate approach. Moreover, techno change differs from traditional IT-projects in focusing on the potential for users (Markus 2004). Therefore, using UGS to support the implementation of service systems, especially during the shakedown phase, seems to be a very promising fit. Our research focuses on making the first step to achieve that fit.

2.3 Motivation Concepts

To understand how to create an environment in which users willingly and actively create, share, and use services to solve their problems, we rely on some theoretical findings from psychology. To make users participate in the project and make use of user-generated services, we have to identify their motives. Motives are individual psychological dispositions (Lakhani, Hippel 2003) that are the result of socialization processes (Ebner et al. 2008; Lakhani, Hippel 2003). Some events activate these motives,

which will encourage behavioral change of individuals, resulting in action. One prominent explanatory model is the motive-incentive-activation-behavior model (MIAB) (Rosenstiel 2003). This model states that through the right external incentives and/or the individual's inborn motives (intrinsic), they can activate and change behavior. This distinction of intrinsic and extrinsic motivation is adopted in many psychological findings such as cognitive evaluation theory (Miller et al. 1988), attribution theory (Heider et al. 1958; Heider 1982), or the two-factor theory (Vincent et al. 1960). We adapt this distinction between extrinsic and intrinsic factors to identify potential motivational factors and incentives during our literature review. Our goal is to integrate the concept of activation-supporting components (Leimeister et al. 2009) as a motivational design requirement and design principle for our overall design science research approach.

3 Overall Design Science Research Approach

In order to increase the success of software implementation during the shakedown phase, we follow the design science principles according to Hevner et al. (2004). According to Hevner et al. (2004) a key characteristic of the design science approach is that it "seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts" (Hevner et al., p.75) and "solving a problem" (Baskerville, Pries-Heje 2010, p. 280).

Figure 1 illustrates our complete design science research approach for our research project adapted from the general design science research methodology (Vaishnavi, Kuechler 2004) and its interpretation following Meth et al. (2015). During our first research iteration, we start with a literature review, which we assess afterwards through a lightweight workshop with 4 experts, each having software implementation and change management experience to discuss possibly relevant tentative design requirements, while simultaneously planning several interviews within two organizations that face techno change challenges, addressing a range of stakeholders from management, external consultants and most importantly, end-users. We focus on software implementation problems and existing motivational and incentive-driven initiatives. The ultimate outcomes are therefore twofold: First, *motivational design requirements* are one result. Second, we create design requirements specifically for UGS, which address the organizational demand for stronger user involvement during shakedown. Based on these findings, we conceptualize a solution during the suggestion phase and create motivational as well as UGS-specific design principles. Next, we plan to prototype UGS based on both motivational as well as context-specific design principles. We realize this by tools and a platform that enables the creation and consumption of learning services in order to share knowledge for specific problems. Their description will then be articulated in design features and describe a service system on a more granular level, similar to the research design of Meth et al. (2015).

We will instantiate the results in form of activation supporting components and pilot a prototype in two real-world scenarios. We separate the resulting interviews and survey as an extra phase, only then can we distinguish between a demonstration phase and an evaluation phase as suggested by Peffers et al. (2008). Iteration II shows discrepancies because we plan to pilot our prototype in two different industries, one being an enterprise system company and the other being an automotive company. We separate the domain-specific design requirements and design principles from the domain-independent ones. This helps us gain more knowledge that is generalizable. Ultimately, through constant abstraction cycles, as Meth et al. (2015) have demonstrated, we will create a blueprint for a motivational service design method for service systems that integrate UGS in the context of shakedown. Due to the nature of UGS, we believe that motivational components are integral and hence constitute them as independent design artifacts. That is the reasoning behind two types of design requirements and design principles and connects us with our brief literature review on existing motivational factors and incentives as a first step toward creating the motivational design requirements.

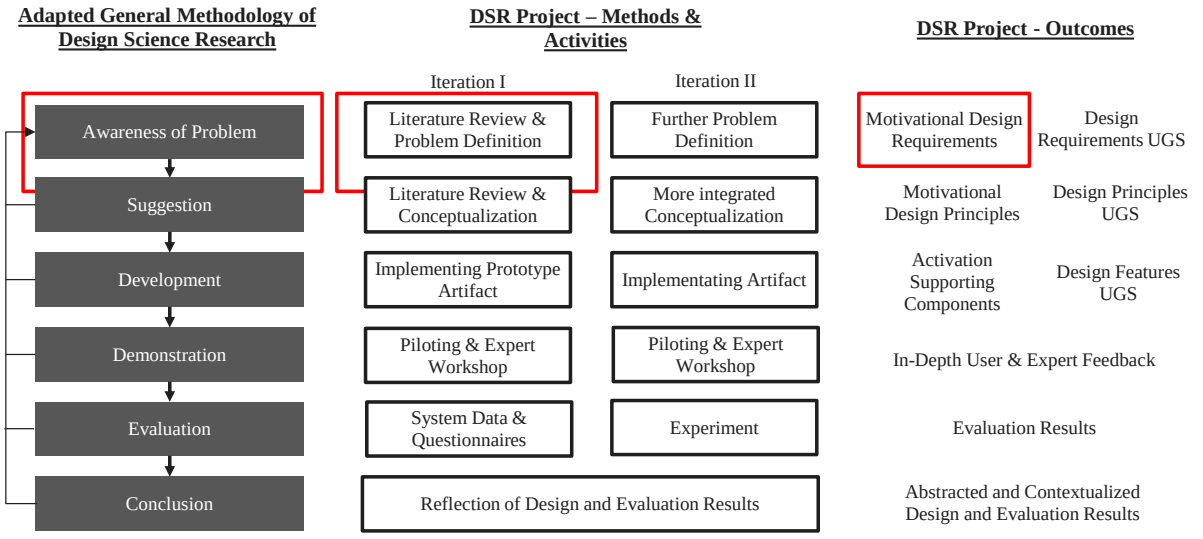

Figure 1 Adapted design science research approach from (Vaishnavi & Kuechler 2004; Meth et al. 2015)

4 Review on Motivational Factors and Analysis

4.1 Method

In order to create motivational design requirements, we conduct a brief literature review guided by Webster & Watson (2002). We start our literature review based on following databanks: ACM Digital Library, ScienceDirect, EBSCOhost, IEEE Xplore Digital Library, SIAM, and Google Scholar using the keyword (("change management" OR "software launch" OR "project launch" OR "project implementation") AND ("participation" OR "activation" OR "motivation" OR "incentives")). The total results were 2314 potential articles. Then we select papers based on their title and remove duplicates. Next, we screen the abstracts and excluded papers that do not seem to address motivational factors or incentives. As a final step, we read all 157 articles to identify, which of them include motivational elements or incentives, resulting in 26 papers. Through forward and backward search, we also included studies from the open source and idea community.

4.2 Preliminary Motivational Design Requirements

In order to create motivational design requirements, we base our reasoning on the MIAB Model (von Rosenstiel 2003). We therefore adhered to two groups of motivational types that are prominent in psychology: intrinsic and extrinsic motivation (von Rosenstiel 2003; Heider 1958) and motives. As table 1 illustrates, we have identified monetary and non-monetary incentives, as well as three prominent intrinsic motives. This table is an isomorphic catalogue, on which future researchers and developers may base their selection of activation-supporting components. Our most prominent finding is the importance of recognition among peers. In open innovation (Leimeister et al. 2009; Gourova, Toteva 2011), as well as in open source communities (Lakhani, Wolf 2005; Jeppesen, Frederiksen 2006; Ghosh et al. 2002), actors crave the recognition of their peers. Therefore, our first design requirement for the design of a service system that enables user-generated services is as follows: **DR1** - *A service system with user-generated services needs to include activation supporting components that enable recognition capabilities for its actors.*

A non-materialistic incentive is characterized by solving personal needs, e.g. when actors face a problem and they have no choice but to solve it in order to continue their work. We therefore argue that solving problems of individuals occurring during shakedown is aligned to the techno change focus on

user benefits (Markus 2004). This further strengthens our idea of user-generated services through the identification of problems and providing solutions for the individuals. We therefore propose following design requirements: **DR2** - *A service system with user-generated services has to provide real solutions for real problems that users actually face (during shakedown).*

The most prominent materialistic incentives are cash bonuses. Most literature concludes that financial incentives play a key role in setting incentives and hence proposes following design requirements: **DR3** - *In order to involve users, the service system that enables user-generated services has to address a set of motives through monetary incentives, e.g. monetary bonuses.*

		Hackman & Oldham (1979)	Klipper & Sandner (2011)	Locke (1978)	Locke & Latham (2004)	Ozimec & Lišanin (2011)	Przygodda (2005)	Wang & Chen (2006)	Weiugo & Yanchun (2010)	Herzberg (1968)	Lawler (1969)	Locke (1968)	Latham (2000)	Latham & Pindner (2005)	Maslow (1943)	Leimeister et al. (2009)	Hars Ou (2001)	Jeppesen, Frederiksen (2006)	Lakhani, Wolf (2005)	Ampofo-Boateng et al. (1997)	Auer, Dobler (2000)	Cheng, Vassileva (2005)	Buregohain et al (2003)	Shah (2005)	Ghosh et al. (2002)	Gourova, Toteva (2011)	Lang et al. (2002)	SUM
motives	Self Realization and Learning	x						x		x						x		x										**5**
	Recognition					x	x			x		x				x	x	x	x	x	x			x	x	x	x	**15**
	Task-Significance	x									x			x		x												**4**
non-materialistic incentives	Community Belonging															x								x	x			**3**
	Solving real problems					x	x	x	x	x		x				x			x			x			x	x	x	**12**
	Helping and sharing knowledge															x		x							x			**3**
	Competition		x						x			x				x												**4**
	Goal-Setting			x					x		x	x	x	x	x						x					x		**10**
	Self-Assessment	x			x			x	x		x			x												x		**7**
	Career chances					x	x		x	x						x		x							x	x	x		**9**
	Equality													x							x		x						**3**
materialistic incentives	Goodies					x																				x		**2**
	Salary									x		x														x		**3**
	cash bonus					x	x								x	x		x	x	x		x			x	x	x	**11**
	profit stakes						x																		x	x		**3**
	social benefits						x		x	x																x		**4**

Table 1 - Identified motives and incentives in current software implementation literature

5 Discussion and Future Work

Before discussing our results, we would like to point out several limitations. First is our limited choice of literature, as we have primarily focused on the software implementation literature and what motivational literature they employ. Second, we acknowledge the large tradition of motivation in IS and have only focused on the software implementation subset, thus only having a limited view, which includes the literature on compensation and rewards. This, however, also reflects the still missing appropriate research on motivation during the chartering phase of software implementation. Third, we did not delve deeper into the vast body of knowledge of change management, as our initial focus was on individual benefits. All three aspects are part of our future work and our next iteration.

In our initial next step, we plan to pilot a service system that enables user-generated services in two organizations and that addresses the shakedown challenge of IT use with different variations of DR1-DR3. Then, we plan to conduct further in-depth interviews at different intervals as hermeneutic cycles to adapt our artifact. At the end of our three-year cooperation, we will evaluate the perceived usefulness of the features that address DR1-DR3 based on a yet to be developed survey. Another stream of research might be the systematic design of these service systems. Existing approaches often lack clearly described phases (Menschner et al. 2011) or are limited to a service – not a service system – perspective (Peters & Leimeister 2013).

Our theoretical contribution lies in demonstrating the fit of techno change and UGS as well as proposing preliminary design requirements for service systems that integrate UGS. From a practitioner's perspective, IT project stakeholders and consultants should consider DR 1-3 when deciding to employ UGS for their holistic techno change projects.

Acknowledgements

The research presented in this paper was partially funded by the German Federal Ministry of Education and Research in course of the project ExTEND (http://projekt-extend.de/), FKZ 01FJ15127.

References

Adolph, S., Kruchten, P., & Hall, W. (2012). Reconciling perspectives: A grounded theory of how people manage the process of software development. Journal of Systems and Software, 85(6), 1269-1286.

Meins: Alter, S. (2012). Metamodel for service analysis and design based on an operational view of service and service systems. *Service Science*, *4*(3), 218-235.

Baskerville, R., & Pries-Heje, J. (2010). Explanatory design theory. *Business & Information Systems Engineering*, *2*(5), 271-282.

Böhmann, T., Leimeister, J. M., & Möslein, K. (2014). Service-Systems-Engineering. *Wirtschaftsinformatik*, *56*(2), 83-90.

Böhmann, T., Leimeister, J.M., & Möslein, K. (2014). Service Systems Engineering. *Business & Information Systems Engineering*, 6(2), 73–79.

Bullinger, H. J., & Scheer, A. W. (2003). *Service engineering—entwicklung und gestaltung innovativer dienstleistungen* (pp. 3-17). Springer Berlin Heidelberg.

Chandler, J. D., & Lusch, R. F. (2015). Service Systems: A Broadened Framework and Research Agenda on Value Propositions, Engagement, and Service Experience. *Journal of Service Research* 18(1), 6–22.

Ebner, W., Leimeister, M., Bretschneider, U., & Krcmar, H. (2008). Leveraging the wisdom of crowds: Designing an IT-supported ideas competition for an ERP software company. *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual* (pp. 417-417). IEEE.

Ernst, S. J., Janson, A., Li, M., Peters, C., & Söllner, M. (2016). IT-Kulturkonfliktheorie und ihre Bedeutung für erfolgreiches Service Systems Engineering-Fallstudie eines Einführungsprojektes für mobile Lernanwendungen in China. Multikonferenz Wirtschaftsinformatik (MKWI).

Ghosh, R. A., Glott, R., Krieger, B., & Robles, G. (2002). Free/libre and open source software: Survey and study. International Institute of Infonomics, University of Maastricht.

Gourova, E., & Toteva, K. (2011). Raising creativity and participation in innovation and knowledge management activities. In *Concurrent Enterprising (ICE), 2011 17th International Conference on* (pp. 1-14). IEEE.

Hars, A., & Ou, S. (2001, January). Working for free? Motivations of participating in open source projects. In *System Sciences, 2001. Proceedings of the 34th Annual Hawaii International Conference on* (pp. 1-9). IEEE.

Heider, F. (1958). The psychology of interpersonal relations. John Wiley & Sons, Inc. New York, NY, US.

Hevner, A. R.; March, Salvatore T.; Park, Jinsoo; Ram, Sudha (2004): Design Science in Information Systems Research. *MIS Quarterly* 28(1), pp. 75–105.

Highsmith, J., & Cockburn, A. (2001). Agile software development: The business of innovation. *Computer*, *34*(9), 120-127.

Jackson, S., & Philip, G. (2010). A techno-cultural emergence perspective on the management of techno-change. *International Journal of Information Management*, *30*(5), 445-456.

Jeppesen, L. B., & Frederiksen, L. (2006). Why do users contribute to firm-hosted user communities? The case of computer-controlled music instruments. *Organization science*, *17*(1), 45-63.

Kim, Y., Lee, Z., & Gosain, S. (2005). Impediments to successful ERP implementation process. *Business process management journal*, *11*(2), 158-170.

Meins: Lakhani, K., & Wolf, R. G. (2003). Why hackers do what they do: Understanding motivation and effort in free/open source software projects.

Lakhani, K. R., & Von Hippel, E. (2003). How open source software works: "free" user-to-user assistance. *Research policy*, *32*(6), 923-943.

Leimeister, J. M., Huber, M., Bretschneider, U., & Krcmar, H. (2009). Leveraging crowdsourcing: activation-supporting components for IT-based ideas competition. *Journal of management information systems*, *26*(1), 197-224.

Lusch, R. F., Vargo, S. L., & Wessels, G. (2008). Toward a conceptual foundation for service science: Contributions from service-dominant logic. *IBM systems journal*, *47*(1), 5-14.

Maglio, P. P., Vargo, S. L., Caswell, N., & Spohrer, J. (2009). The service system is the basic abstraction of service science. *Information Systems and e-business Management*, *7*(4), 395-406.

Maglio, P. P., & Spohrer, J. (2008). Fundamentals of service science. *Journal of the Academy of Marketing Science*, *36*(1), 18-20.

Markus, M. L. (2004). Techno change management: using IT to drive organizational change. *Journal of Information technology*, *19*(1), 4-20.

Majchrzak, A. (1992). Management of Technological and Organizational Change. In: G. Salvendy (ED.) Handbook of Industrial Engineering, 2nd Edition, NY: John Wiley, 767-797.

Meth, H., Mueller, B., & Maedche, A. (2015). Designing a Requirement Mining System. *Journal of the Association for Information Systems*, *16*(9), 799.

Müller, R. M., & Thoring, K. (2012). Design thinking vs. lean startup: A comparison of two user-driven innovation strategies. *LEADING THROUGH DESIGN*, *151*.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2008). A design science research methodology for information systems research. *Journal of management information systems*, *24*(3), 45-77.

Peters, Christoph; Blohm, Ivo & Leimeister, Jan Marco (2015): Anatomy of Successful Business Models for Complex Services. Insights from the Telemedicine Field. *Journal of management information systems*, 32(3), 75-104.

Peters, Christoph & Leimeister, J. M. (2013): TMÂ³ - A Modularization Method For Telemedical Services: Design And Evaluation. *Proceedings of the 21st European Conference on Information Systems (ECIS)*, Utrecht, Netherlands.

Roberts, B., Jarvenpaa, S., & Baxley, C. (2003). Evolving at the Speed of Change: Mastering change readiness at Motorola's semiconductor products sector. *MIS Quarterly Executive*, *2*(2), 58-73.

Silva, L., & Hirschheim, R. (2013). Fighting against Windmills: Strategic Information Systems and Organizational Deep Structures. Management Information Systems Quarterly, 31(2), 327–354.

Soh, C., & Markus, M. L. (1995). How IT creates business value: a process theory synthesis. *ICIS 1995 Proceedings*, 29–41.

Surowiecki, J. (2006). The Wisdom of Crowds. Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economics, Societies and Nations. *Personnel Psychology* 59 (4), 982–985.

Kuechler, B., & Vaishnavi, V. (2004). Design Science Research in Information Systems.

Vandaie, R. (2008). The role of organizational knowledge management in successful ERP implementation projects. *Knowledge-Based Systems*, *21*(8), 920-926.

Vargo, S. L., & Lusch, R. F. (2004). Evolving to a new dominant logic for marketing. *Journal of marketing*, *68*(1), 1-17.

Rosenstiel, L. V. (2003). Grundlagen der Organisationspsychologie-Basiswissen und Anwendungshinweise. 5. überarbeitete Auflage. *Stuttgart: Schaeffer Poeschel*.

Wagner, C. (2004). Wiki: A technology for conversational knowledge management and group collaboration. *The Communications of the Association for Information Systems*, *13*(1), 58.

Webster, J., & Watson, R. T. (2002). Analyzing the past to prepare for the future: Writing a. *MIS quarterly*, *26*(2), 13-23.

Zmud, R. W., & Cox, J. F. (1979). The implementation process: A change approach. *MIS Quarterly*, 35-43.