Requirements for an IT Support System based on Hybrid Intelligence

Simon L. Schmidt
University of Kassel
simon.schmidt@uni-kassel.de

Mahei Manhai Li
University of Kassel
mahei.li@uni-kassel.de

Christoph Peters
University of St. Gallen; University of Kassel
christoph.peters@unisg.ch

Abstract

In our digital world, all companies need IT support. IT support staff are under increasing pressure solving increasingly heterogeneous user tickets. Hybrid intelligence could solve many issues due to the combination of machine power and individual strengths of humans. As a part of a bigger design science research project, this paper derives requirements for an IT support system based on hybrid intelligence (ISSHI): In total 17 literature-based requirements and 21 interview-based requirements with IT-support managers and support agents from three different companies were consolidated into 24 requirements to inform an ISSHI system architecture. This architecture serves as a foundation for future research regarding hybrid intelligence in IT support.

1. Introduction

Tuesday morning, 8 AM, the shift of Nik, a 1st level IT support employee (also called a support-agent) begins. The first incoming requests of the day have already automatically been solved by his peer: a chatbot. As his first request of the day, he receives a type of problem that is new to him, and he does not know how to solve it. Fortunately, he has a new IT support system based on hybrid intelligence (ISSHI) in place that recommends him solution materials based on the request at hand. Whereas previously he would have forwarded this request to his more experienced colleagues, today he can solve the problem on his own with the help of the suggested solution material.

To create this new setting for IT support, and to go beyond the described scenario, this paper develops requirements for an ISSHI with the goal that the fictive description above shall not remain a dream of the future but become reality by combining the human and artificial intelligence (AI) to make use of the individual strengths of each actor in a hybrid intelligence (HI)-based approach. The individual strengths of AI in IT support, such as retrieving knowledge based on the current request [2, 3] and routing tickets to the corresponding expert [4, 5], will be combined with the strengths of humans, such as adaptability to new data, solution-seeking capabilities, and creativity. Further, by using a HI-approach each

individual will achieve higher performance compared to the individual performance and can learn from another [6, 7]. The system will become better during usage. However, there are no studies to date that use a holistic perspective on designing an ISSHI. That means, that this paper considers all stakeholders, tasks, organizations, and the interactions between them. Hence, it is worthwhile to consider how IT support is typically performed to date, especially under the prevalence of the difference between an ISSHI and a non-HI-based IT support:

(1) A user encounters a problem, and s/he (2) contacts the IT support via the communication channels such as telephone, online chat, or email. (3) A trained support-agent receives the ticket and records it in a ticketing system. S/he (4) continuously documents the progress in the system for transparency to the users and internal coordination and traceability of the issue. Hence, tickets contain helpful documentations when support-agents face the exact same problem. The next step (5) checks the ticket for completeness (e.g., PC-nr., operating system, etc.). If information is missing (6) the support-agent communicates with the user. Afterward, the support-agent (7) prioritizes the ticket based on potential impact and urgency. With the help of a category tree, the ticket (8) gets categorized. This category is often used to (9) route the ticket to a specialized support-agent/ department. The issue is then (10) checked for a solution. If no immediate solution is possible, (11) solution material is sought. This can be found in a knowledge base (e.g., an internal wiki), a previously solved ticket (support-agents document the solution attempts within these tickets), the onboarding material, or personal documentations. Peer support also plays a major role, especially through shared chats. Support-agents often exchange solutions and advice, forming a small knowledge base within these chats. (12) If solution material was found and the ticket is solved, the user is informed about the solution and the ticket is closed. (13) If no solution was possible, the ticket is escalated to a higher level.

Given this context, today`s IT support can be characterized by the prevalence of three major challenges: (1) Support-agents are spending too much time on trivial tasks and consequently do not have

enough resources to properly solve more complex tickets. Here, self-created chatbots can relieve the human workload for recurring tasks. Additionally, HI-based categorization, prioritization, translation, and routing of tickets can increase efficiency. (2) IT support tasks are knowledge-intensive services in which support-agents need to rely on many years of experience to solve the often highly complicated issues of the users. This makes training new IT support employees time-consuming and adds additional workload to the seasoned support-agents. Hence, all support-agents can benefit from an automatic matching of solution materials to the current request because it is time consuming to manually search through millions of possible solution materials. (3) There is immense knowledge externalized within the previously solved tickets, own documentations, chats, and knowledge base articles. However, this knowledge is difficult to access because it is often outdated or does not match the problem at hand. Hence, all sources of knowledge should be combined.

This research paper will focus on deriving requirements for an ISSHI regarding the described process above. This complex task is based on various factors such as the as-is process, data management, algorithm development and improvement, and the individual requirements of IT support departments. Hence, to create such an ISSHI, we ask the following research question: *What are the technical requirements for an ISSHI?*

To answer this question, the paper is structured as follows. First, the research approach is described. Second, we provide an overview of relevant IT support literature, introduce the concept of HI and IT service management (ITSM), and derive requirements from the literature. Third, we present the results of 24 expert interviews to derive requirements from practice. Fourth, the derived requirements from literature and practice are combined. Fifth, the requirements are evaluated based on 5 expert interviews. Sixth, an architecture is designed based on the requirements. The paper concludes with a discussion, limitations, implications for future research, and a conclusion.

2. Research design

This paper is based within a larger design science research (DSR) project and starts by deriving the requirements for an ISSHI. We are accompanying the development of the ISSHI with three companies, two small and medium-sized enterprises, and a global manufacturing and service company. The former two companies offer professional IT support to their customers, while the global company directs its global support internally. Through the different perspectives, the gained knowledge can be applied to IT support in general.

Following the DSR approach from Hevner [1], this paper contributes to the existing body of knowledge by (1) formulating the problem state of the IT support and deriving a holistic set of requirements for the IT support based on literature and practice. Hence, the requirements elicitation is based on the combination of the most frequently mentioned requirements from practice and (2) relevant literature for the interdisciplinary usage of HI in IT support, such as HI, AI, ITSM, statistics, data management, information retrieval, and knowledge management. The main goal was to identify the key requirements from the literature. Hence, we asked IT support experts and other researchers about their paper recommendations and used forward and backward search to screen for suitability based on abstracts to find issues and requirements for HI-based systems and IT support in general. (3) 29 qualitative semi-structured in-depth interviews [8] in two interview series were conducted to derive, evaluate, and refine the requirements from the practical point of view. Hence, the interview partners were not confronted with requirements from the literature and were asked questions about their key challenges/ requirements in IT support. All requirements were recorded, and further details were gained during the interview. If more than three IT support experts have stated the same requirement, they were used for this paper. We selected potential interview partners based on their expertise on the subject matter of IT support and tried to cover both decision-makers and support-agents. Consequently, we interviewed 24 stakeholders related to IT support from the three companies that take part within our larger DSR project, including general managers, knowledge managers, and support-agents from all levels (1st level, 2nd level, 3rd level, and on-site support). We conducted two interviews on-site and the rest online via Microsoft Teams. The average duration of the interviews was approximately 50 minutes. We stopped our initial interview series after conceptual saturation when we did not receive new requirements or generated further insights. In step (4) the requirements from the literature and the expert interviews were combined. To (5) evaluate and refine these requirements we started a second interview series with 5 experts (2 managers and 3 support-agents). The duration of the second interview series was between 37 and 52 minutes. The evaluation criteria were applicability and suitability of the requirements in IT support regarding the integration of HI within the IT support ecosystem, understandability, correctness, and completeness of the requirements. The final (6) documentation of the requirements and the (7) design of an architecture for ISSHI to leverage the IT support with HI and categorize all requirements

within one figure are the last steps of this paper. The (8) instantiation of the requirements and the evaluation of the instantiated requirements will be future steps.

The objective of this paper is to identify, analyze, evaluate, and refine novel requirements within the domain of IT support regarding HI. Hence, the artifact will be a set of requirements, which will be categorized and within an architecture to help scholars and practitioners to build the personalized design features of an ISSHI. According to the IS success literature [9], there exist six success dimensions. These dimensions are considered in this paper to later describe a successful collaboration between humans and AI after deploying and using the system. Hence, the requirements will already be formulated to fit the constructs of the model e.g., system quality (the overall quality of ISSHI after instantiating the requirements) and information quality (the quality of the data that is used by the HI). Following, this paper can be seen as one of the "landmark papers that inform the development of alternative solutions" [10]. Nunamaker [10] describes these papers as proof-of-concept research that define the class of problems with "generalizable requirements for solutions to the class of problems". Ultimately, this proof-of-concept paper derives requirements for an enhanced understanding of the class of problems within the domain of IT support and HI. Further, the results can be seen as an exaptation [11] because possible solutions of HI, which are often based on some kind of machine learning algorithms, are already well developed. The exaptation is also based on the holistic view of derived new requirements and the approach to use HI in the domain of IT support. Hence, we argue, that the application domain maturity is rather low, while the solution maturity of single components from other fields is high. Finally, the requirements will be used to design a scalable architecture that satisfies all requirements based on rigor and relevance.

3. Deriving requirements from literature

3.1 Hybrid intelligence

All requirements from chapter 3 are indicated by Ln where n is the number of the derived requirement that is based on Literature. HI relies on combining artificial and human intelligence [6, 7] to (a) relieve and (b) support humans, and (c) to achieve higher performance compared to the individual performances of each actor. However, this form of human-machine collaboration is scientifically underrepresented in IT support and thus offers space for exploration.

HI is intended to leverage efficiency potential (e.g., by using the strengths of an AI and humans), leverage the complementary strengths of the IT support staff and machines (e.g., when AI is uncertain about a result and human intelligence is needed), and relieve employees from monotonous work (e.g., for simple tasks in IT support such as asking for a hardware number and a user name) [6, 7]. We formulate the first requirements to relieve support-agents from simple tasks (**L1**), support humans during difficult tasks (**L2**), and achieve higher performance with the system (**L3**).

A prerequisite for a successful HI-based ISSHI is a working AI that performs a predefined task based on a set of valuable data e.g., for supervised machine learning algorithms. Whenever only little data is at hand or data shifts are to happen often, HI approaches are particularly valuable. Here, support-agents can provide feedback on the results of the ISSHI and the algorithm learns. One example of this is the human-in-the-loop approach: AI learns from the human's input; s/he takes the role of a teaching person, and the machine extends its intelligence according to the new annotated labels [7]. Accordingly, the system should provide the ability for the IT support staff to give feedback to the AI (**L4**)

In this paper, two main actors interplay with the system. These are the (1) IT support staff, such as general manager, knowledge manager, and support-agents, and (2) the AI. However, the IT support staff is at the center of consideration in the sense of a high-road strategy. Actors that are out of scope are (3) the users that ask for help when an error occurs and (4) third-party developers that can extend the provided services to the system. Due to the interaction of the actors, the ISSHI can be seen as a socio-technical system, and techniques from service systems engineering can be applied [12].

To design the ISSHI as potentially scalable and useable for a variety of companies, application programming interfaces (API) can be used. They can also improve architectural efficiency and increase customer traffic [13]. By providing an API, IT support departments can use offered services of the system, build their personal frontend on the backend, or integrate the services within their existing IT support system. The system can also benefit from APIs when it serves as a digital platform and enables network market effects by facilitating the interaction between the companies, the platform owner, and third-party developers [14, 15]. Hence, we formulate the requirement to use the API-first paradigm when designing the system (**L5**).

3.2 IT-service management

ITSM are guidelines and practices to implement, deliver, and manage IT services as a connector between user and IT support [16]. An IT service can be anything that an organization supplies support for, such as hardware (e.g. notebooks, or servers), software

(e.g.: MS Teams), or cloud solutions (e.g. computing power-as-a-service). Hence, the goal of ITSM is to deploy, operate, manage, maintain, and optimize IT services with processual, reactive, or proactive approaches. ITSM can be seen as a bundle of best practices. However, in this paper, we will focus primarily on HI-based approaches to support-agents during day-to-day operations, reduce system downtime, and increase the availability of IT services. Widespread ITSM frameworks are the IT Infrastructure Library (ITIL) and Microsoft Operations Framework (MOF) [17]. In 2019, ITIL 4 was launched, and includes 9 principles, integrating agile principles and cultural embeddedness of IT across business units and customer feedback. The fourth version aims to make ITSM more adaptable, customizable, and flexible.

ITSM can lead to (1) improved customer satisfaction [18], (2) higher service quality [18, 19], and (3) higher efficiency due to standardization and automation of processes [19]. Ultimately, the service system should support the automation of processes (**L6**) and the standardization of processes (**L7**).

3.3 HI & AI in the context of IT support

This section derives requirements based on literature at the intersection between statistics, data management, ITSM, software engineering, AI, and HI. This study differs from other studies by providing a broad perspective of requirements for a system for IT support focusing especially on HI. As most of the following requirements are derived from non-HI-based literature it will be transformed according to the specific use case when combining relevance and rigor in chapter five.

The first requirement for the system is the implementation of matching algorithms to recommend potential solution material. With techniques from information retrieval, natural language processing, and data mining support-agents can retrieve knowledge from previously solved tickets (**L8**) or from a knowledge base (**L9**) based on their current ticket [2, 3]. This can shorten the processing time to find a proper solution to the underlying issue at hand.

By implementing the automatic classification of tickets, several issues can be resolved. First, the classification based on a category tree can be automated (**L10**) and support-agents are relieved from classification work, which is currently a manual process. This classification can be done with text mining techniques [20], e.g., Naïve Bayes or support vector machine text classification algorithms. Besides the reduced human effort, the automated classification can decrease the resolution time, wrong classifications, and can increase the accuracy of the matching algorithms when combined with the resolutions of the classification algorithms.

The routing to a team or specific support-agent is often based on the classification of the ticket but can also be based on specific written algorithms for this specific task (**L11**). Wrong assignments and re-routing tickets are thereby prevented and the time to fix an issue can be decreased [4, 5]. This can either be done based on pre-defined classes, the ticket history, or social network models.

Data quality is of utmost importance when it comes to services that are based on data. Whenever solution material is retrieved from previously solved tickets, they need to have high quality to help solve an issue (**L12**) [3]. This can be achieved by detecting low-quality textual data and separating it from the rest. These techniques have been proven for product reviews [21], crowdsourcing texts [22], and IT support tickets [3]. A similar approach besides the textual quality can be to assess the helpfulness [23] of a ticket. All these approaches can benefit from HI with continuous feedback from support-agents during the operation of the system to train the algorithms.

Chatbots (and voice bots) can interact as the first touchpoint with users in the IT support via natural language interfaces [24, 25]. Increasing productivity is one of the most frequently reported motivations to use chatbots [26]. Hence, automizing tasks, especially low-skilled tasks (**L13**), can (a) relieve IT support staff from ongoing questions and (b) save money. A chatbot also has 24/7 availability. However, several requirements exist for the integration of a good chatbot to the ISSHI. First, a chatbot needs good data at hand to (a) understand the users and (b) answer appropriately (**L14**) [27]. Therefore, the data needs to be (a) easily integrated or (b) automatically integrated, e.g., with APIs (**L15**) [27]. Second, creating a chatbot needs to be fast and easy (**L16**), which can be achieved if a no-code or low-code platform is used that allows everyone with no programming skills to develop a chatbot. Third, because the interaction with the IT support is both emotional and factual [25], sentiment detection should be integrated (**L17**).

4. Deriving requirements from interviews

Through the interviews, multiple requirements became apparent that are in line with the literature. All requirements from chapter 4 are indicated by En where n is the number of the derived requirements that are based on the expert interviews. Here, we present the derived requirements that were approved by both, support-agents, and management, with a minimum of three people mentioning them.

An important factor to solve requests is the know-how of support-agents. It became apparent that specific support-agents specialized in a certain topic.

Due to this circumstance, the IT support is dependent on this person, leading to issues as soon as s/he becomes sick or leaves the company. Hence, an ISSHI should prevent accumulating specific knowledge centrally with one person (**E1**).

Expert knowledge can only be gained if experienced support-agents share their knowledge efficiently. However, there are many different systems where support-agents can store knowledge, e.g., chats, wikis, tickets, etc. The transfer of knowledge between support-agents is often complicated because of too many platforms and databases. To prevent the loss of knowledge, the system should provide APIs to centralize all sources of information in one central database (**E2**). A widely used source is the knowledge base in the form of wikis. However, the maintenance of these knowledge base articles is often time-consuming due to updates to new software versions or the rollout of completely new hard-/ software. Hence, the database of the system with its different sources should be easy to update or should be updated automatically (**E3**). Based on this database, to help especially new support-agents gain knowledge, potential solution material should be provided to them (**E4**). Ultimately, by providing them with the solution materials they can increase their knowledge and experience and ultimately increase their productivity. Hence, the system should match the solution material automatically based on the current request (**E5**) due to the enormous time pressure of the support-agents.

Nevertheless, when solution material is provided by previously solved tickets, the documentation of the tickets is of utmost importance. Following this, the system should guide support-agents during the process of documentation for the understandability and comprehensibility of the solution (**E6**). The completeness of the information is a major property of a well-documented ticket. However, receiving information from users can be very time-consuming. To relieve support-agents from the task of manually completing information and checking with users, a chatbot could be used. This chatbot asks users about missing information (**E7**) and thus saves time and prevents incomplete documentation. Nevertheless, incomplete, or poor solution materials could still arise and already exist. Hence, an ISSHI should detect low-quality solution material that is already stored/ will be stored in the database (**E8**) and should separate it from the high-quality solution material (**E9**).

Whenever support-agents open a solution material, they struggle to identify the relevant parts such as step-by-step solution, problem description, failed attempts to resolve the issue, or description of general data (operating system, software version, etc.) because support-agents document their tickets differently. For faster identification of the needed sub-part of a solution material, the system should automatically annotate these parts (**E10**) as well as provide support-agents with the ability to manually (re-)annotate them (**E11**).

To further improve the efficiency in understanding the main issue of incoming requests, support-agents asked for automatic summaries whenever they receive new tickets. Therefore, the system should automatically provide summaries of requests (**E12**) for the support-agents.

To use potentials in automizing the process and to support support-agents in the early stage of solving and documenting a ticket, the system should provide automatic categorizing (**E13**) and prioritizing (**E14**) of incoming requests. Following the categorization, the system should also provide automatic routing to correct specialized teams/ support-agents that are often based on this category of the ticket (**E15**). This requirement especially arises as soon as major bugs or problems occur that a support-agent is unable to solve. In these cases, the system should recognize such major problems and should route them automatically to the development team (**E16**).

Since many IT support departments are outsourced, and/or the company offers multiple language support, support-agents often speak different languages compared to their peers and customers. Hence, the system should automatically translate all sources, such as solution material and tickets, that support-agents/ users interact with to their native language (**E17**).

As with almost all AI-based projects the rollout of such an ISSHI should support the staff instead of replacing them (**E18**) due to acceptance of the system. Hence, the IT support staff should be empowered to decide themselves which processes are to be automatized with chatbots because they know the recurring and time-consuming tasks. Two requirements arise: First, the staff should be able to decide where to use chatbots (**E19**). Second, the staff needs to learn how to program chatbots (**E20**), which also changes the job profile. Additionally, we noticed that support-agents prefer to use the same process instead of learning a new one. Hence, the system should consider the individual official and unofficial processes of the IT support department (**E21**). This requirement should be applied during all phases of designing, piloting, and integrating the system within the IT support ecosystem.

5. Requirements for an IT support system based on hybrid intelligence

Based on relevance and rigor, the requirements of the system are consolidated and presented in this

chapter. These requirements show what a HI-based ISSHI requires based on literature and practice. Hence, the presented requirements in chapters three and four are combined and can be seen Figure 1. However, these requirements should be implemented differently depending on factors such as corporate culture, available resources, management, and tech affinity.

One of the main goals of the introduction of an ISSHI is to relieve support-agents from simple tasks. Hence, the system should provide automatization of processes to relieve support-agents from simple tasks (**R1**). Here, the benefits of a HI-based approach can be seen because support-agents choose which tasks to automize while still relying on their main competencies in solving complex and new tasks. Nevertheless, the system should also provide the IT support staff during hard tasks with help (**R2**). By combining the analytical strengths of humans and the capacity to compute loads of information with an AI the hard tasks can become easier. However, the AI may not deliver perfect results by default and data shifts are likely to happen. Following, the system should provide continuous support-agent-based feedback to the AI (**R3**) to learn and to adapt to new circumstances; the teaching-interaction between the different actors human and machine is a central component of the system. To train the AI and to make use of all possible sources of knowledge the system should be provided with APIs to support the integration of different sources of data, and to support manual/ automatic updating of data (**R4**). Chatbots can increase the overall productivity of IT support and are available 24/7. However, to build these chatbots, support-agents should be provided a tool to easily build them (**R5**) e.g., based on a no-code/ low-code platform. As the chatbots can only be as good as the data they rely on they should be provided with high-quality data only (**R6**). Following, the built chatbots should be updated regularly by the support-agents regarding the ability to solve the issue they are supposed to solve. Further, the support-agents should decide themselves where to use a chatbot (**R7**) because (1) they can decide best which tasks are reoccurring and (2) humans are in the center of observation within the HI-based system. Due to the high emotionality of users when they face errors, the chatbots should be provided with sentiment analysis (**R8**) to interact emotionally correct with the users. This analysis can be constantly improved with feedback from support-agents and users to the AI. A specific ability of chatbots over all three companies is to collect missing user information (**R9**) because it is very time-consuming when done manually by the support-agents. Two possibilities arise: (1) The system automatically detects missing information and the chatbot immediately contacts the user or (2) a support-agent detects missing information and delegates the task to collect them. By manually triggering this process the system learns to detect the missing information automatically better. A general requirement that applies to the integration of chatbots, giving feedback to the system, and integrating the system within the ecosystem is to design the processes standardized and in line with the as-is processes (**R10**). Following, support-agents should be provided with help of the system wherever possible e.g., by retrieving knowledge from all sources that are available within the company (**R11**) such as knowledge base articles, wikis, chats with peers, product documentation, etc. As soon as these knowledge resources are combined within one database support-agents should be provided with retrieved solution material based on the current request they work on (**R12**) to support them during the time-consuming process of searching for relevant solution material. Such a HI-based matching algorithm benefits from feedback e.g., by evaluating the quality of the match regarding the correctness of the topic from the retrieved solution material. To support support-agents further, the system should provide the category (**R13**) and the priority (**R14**) of a ticket automatically. Here, HI will be used to relieve support-agents from monotonous tasks and to teach the AI when a category/ priority was predicted wrong by manually setting the category/ priority correct. Tickets are often based on the category and the priority. Hence, the system should also provide automatic routing of tickets (**R15**). This increases the first-time fix rate as well as the duration until an issue is fixed because (1) the routing happens immediately after a new ticket is created, (2) bugs will be sent to a development team instead of a support-agent, and (3) re-routing tickets will be decreased. However, if the ticket was routed wrongly by the AI, an experienced support-agent will correctly route the ticket and the AI learns. Following, wrongly routed tickets will decrease with the time the system is in use. Data quality is of utmost importance when using HI-based systems thus the data should continuously be updated and maintained while only containing high-quality information (**R16**). Following this, the system should automatically detect low-quality data and separate it from high-quality data. Through the HI-based approach, these quality detection algorithms can be trained based on feedback regarding the information the IT support staff receives e.g., solution material in the form of knowledge base articles and previously solved tickets. To decrease the number of low-quality data, support-agents should be guided during ticket documentation (**R17**) to increase understandability and comprehensibility. To further

increase the quality of tickets, the system should provide them with automatic summaries (**R18**) to understand the content of a ticket briefly. The HI-based system learns from the feedback of the support-agents that evaluate whether the content is summarized correct/wrong e.g., with the feedback form of thumbs up/ down. To identify the relevant parts of solution materials such as step-by-step solutions or problem description, the system should provide them with annotations regarding these sub-parts (**R19**). Here, the AI automatically identifies the corresponding sub-parts and highlights them in colors. If annotations were incomplete or incorrect a support-agent can manually update the annotations and the system learns from the feedback. In other words, the support-agent not only corrects the annotations but also trains the AI all at once. Due to outsourcing and offering multiple language support, support-agents should be provided with automatic translations to their native language regarding all kinds of sources e.g., requests, previously solved tickets, knowledge base articles, etc. (**R20**).

6. Evaluation and refinement

The requirements are based on the problem statements of the 24 in-depth interviews and the literature. We evaluated and refined the requirements with a second interview series with five more interviews. This series can be seen as an evaluation 1 according to [28]. Therefore, the requirements can be considered as the input to the interview series. The output are the justified and revised requirements based on the problem statements of the IT support experts and their additional comments. Applicability, suitability, correctness, understandability, and completeness were the evaluation criteria. In total one requirement was revised, and four requirements were added. Refined parts of the requirements can be found in *italics* together with the added requirements in Figure 1. After presenting the requirements, the HI approach was also presented to the experts which received positive feedback due to putting the IT support staff in the center of consideration. With this point of view support-agents as well as the management thinks that there will be almost no resistance when such an ISSHI is rolled out in the three companies. It became apparent that the combination of artificial and human intelligence is seen as positive because strengths will be strengthened, and the staff can be relieved from monotonous work. However, we identified minor doubts from the management regarding the usefulness during the shakedown phase when the system is rolled out within the companies and the algorithms are about the be trained. Following, during the critical phase of shakedown, the IT support staff needs to give feedback to the AI to learn. By providing the feedback the results will become better and the system will be more powerful resulting in more users. Hence, we added the requirement that the IT support staff should be rewarded when giving feedback to the system (**R21**) e.g., by using techniques from nudging or gamification. This helps twofold: (1) The AI continuously learns and (2) the support-agents are motivated to use the system more often.

Requirements from Literature L and Experts E	Initial requirements with *refinements*
L1, L3, L6, L13, E18	R1: Provide automatization of processes to relieve agents from simple tasks
L2, L3, E18	R2: Provide the IT-Support staff during hard tasks with help
L4	R3: Provide continuous agent-based feedback to the AI
L5, L15, E2, E3, E4	R4: Provide the system with APIs to support an integration of different sources of data, to support manual/ automatic updating of data, *and to integrate third-party services*
L16, E20	R5: Provide a tool to easily build chatbots for the IT-Support
L4, L14	R6: Provide chatbots with high-quality data
E19	R7: Provide agents with the ability to decide where to use chatbots
L4, L17	R8: Provide chatbots with sentiment analysis
E7	R9: Provide chatbots to collect missing user information
L7, E21	R10: Provide system with standardized processes based on the as-is processes
L8, L9, E1	R11: Provide agents with retrieved knowledge from all sources
L4, L8, L9, E5	R12: Provide a matching algorithm to retrieve solution material based on the current request
L4, L10, E13	R13: Provide an automatic categorization of tickets
L4, E14	R14: Provide an automatic prioritization of tickets
L4, L11, E15, E16	R15: Provide an automatic routing of tickets
L4, L12, E8, E9	R16: Provide database only with high-quality information
E6	R17: Provide agents with guidance to document tickets
L4, E12	R18: Provide tickets with automatic summaries
L4, E10, E11	R19: Provide solution materials with annotations of sub-parts
L4, E17	R20: Provide all sources with automatic translations to the native language
Additional requirements	
	R21: Provide rewards for IT-Support staff for giving feedback to the system
	R22: Provide data with pseudonymization according to general data protection regulation
	R23: Provide a cockpit overview to identify knowledge gaps of agents based on routing
	R24: Provide a personalized user interface

Figure 1: Refined requirements

Regarding completeness of the individual requirements the **R4** "Provide the system with APIs to support an integration of different sources of data and support manual/ automatic updating of data" was refined. Support-agents asked for the ability to integrate third-party services such as using a chatbot within Microsoft Teams instead of building a new platform because they experienced that users are more responsive on Microsoft Teams compared to e.g., e-mail and the integrated chat of the ticketing system in use. Hence, the integration of widely used third-party services in IT support was added. This also includes other third-party services beyond chatbots within Microsoft Teams. Here, requirement engineers need to define the individual needs of support-agents within their respective companies.

Whenever sensitive user data, (e.g. home address), are written down within a ticket, the system should also meet the general data protection regulation. As soon as data is stored on databases not owned by the company itself, all sensitive user data should be pseudonymized (**R22**).

Based on a manually triggered routing history of support-agents an IT support manager can identify knowledge gaps of their support-agents. To relieve managers from this manual work, the system should provide them automatically with insights for training based on the routing history of support-agents (**R23**). Finally, managers asked for a personalized user interface for their own company to meet the corporate identity, especially when communicating with external users (**R24**). Overall, the requirements were all understandable and correct for the interview sample and we stopped the series after five interviews.

7. Discussion, limitation, and future work

This study used literature and in-depth interviews for requirement elicitation of an ISSHI. Regarding the retrieved requirements, we designed an architecture of an ISSHI that can be found in Figure 2. It shows how

these requirements, scholars and practitioners can build features for their ISSHI and continuously design their own service architecture. While all the requirements are specifically derived for IT support, they provide deep insights into the design of HI-based systems for similar domains. On the other hand, due to the broad intersection of the presented literature stream, some requirements from the IT support can easily be adopted from related domains after adjusting them to the HI-based approach. Requirements not based on HI are the general requirements (R1, R2, R10), user interface (R24), and API design (R4).

The requirements show that the categories classification (R13), chatbots (R5, R7, R8, R9), data quality (R6, R16, R17, R22), knowledge base (R11, R19) matching (R12), prioritization (R14), routing (R15, R23), summarization (R18), and translation (R20) can be based on HI. Possible services for the
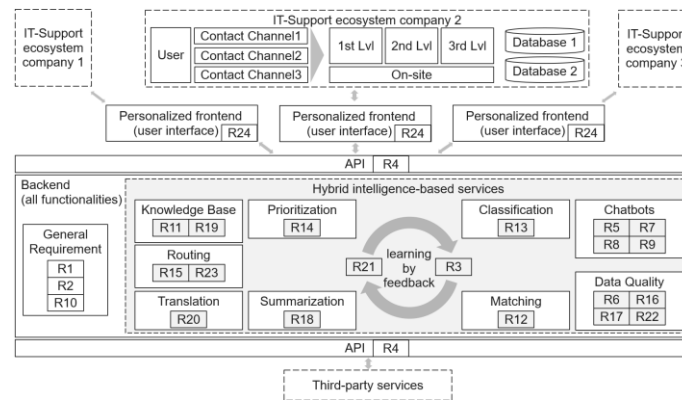

Figure 2 Designed architecture

the individual and existing IT support ecosystems of the companies, which can be seen in the upper part indicated by companies 1 – 3, interact with ISSHI. Within the ecosystem of the individual companies 1 – 3, the user contacts the IT support by different contact channels such as online forms, telephone, or e-mail, and the IT support with its different levels solves the issue. Here, different databases such as knowledge bases or previously solved tickets can be used to find appropriate solution material for the issue at hand.

At all times, to support support-agents, the ISSHI offers its services either (1) via personalized frontend or (2) via backend functionalities directly by the API. Because ISSHI integrates third-party services, the companies do not need to program their own APIs; these services are also callable by the same API of the ISSHI. By calling the API functionalities or by using the user interface, the services of ISSHI are called.

These services are the main functionalities which are divided into several categories. These different categories can be seen within the boxes in the middle part of Figure 2. Here, the boxes describe the grouped requirements from the previous section. Based on

category knowledge base could be (1) combining all possible sources of knowledge such as training documentation, previously solved tickets, and knowledge base articles, to a central point of data to make them easily machine-searchable, and (2) highlighting specific parts within solution materials such as step-by-step solutions and problem description to decrease the cognitive load. Services for the category data quality can be derived from the requirements R6, R16, R17, and R22. Here, machine learning algorithms can be trained to detect low- and high-quality solution materials within all available data. Further, another feature should separate the high- from low-quality data. Consequently, to confirm to the general data protection regulation, all data should be pseudonymized by another service of ISSHI.

To increase the performance of all HI-based services, such as detecting low- and high-quality solution materials and pseudonymizing data, the IT support staff gives feedback (R3, R21) to the system. Learning by feedback is in the center of all HI-based services and is used to increase their performance.

This feedback can be gained in various ways e.g., by correcting wrong predictions or giving direct feedback to predictions. Following this, the system becomes more intelligent, and the results will be better. Ultimately, the system will be used more often due to the increased capabilities which leads to more feedback and enhanced learning of the AI. During the critical shakedown phase, when the system is rolled out and the AI has not gained much feedback, more feedback can be gained if the staff receives awards or is nudged.

By designing this architecture, a new platform for IT support based on HI will be created by ISSHI. Several actors benefit from it: (1) Users due to faster response time and enhanced problem solving of the support-agents. (2) Support-agents due to automatic classification, prioritization, routing, translation of all information to the native language, summarization of tickets, solution material retrieval with annotation of important parts, keeping data quality high, and solving issues automatically by self-programmed chatbots. (3) IT support managers due to increased efficiency of the whole department and recognizing training needs of support-agents automatically. (4) Quality managers due to higher quality of knowledge base articles, and documentations of tickets. Further, (5) different companies can benefit from another by sharing knowledge within non-competitive tasks such as internal IT support for their employees where knowledge can be shared without any risk if the requirements according to general data protection regulation are met.

Future research will look at deriving more requirements based on a systematic literature review and implementing and piloting the ISSHI. We already started with programming the first services and will evaluate them based on the IS success literature in combination with IT support specific evaluation criteria such as first-time fix rate, duration to fix, perceived usefulness, and perceived ease of use. Finally, we plan to evaluate the whole system in action design research. Considering the interplay of different companies and a platform approach, we will explore the possibilities for companies regarding knowledge-sharing in non-competitive tasks and decentralized learning of the HI-based system. Further, different feedback possibilities will be analyzed based on an evaluation of the number and the quality of feedback. Additionally, users will also be in consideration for giving feedback to the AI. We hope to inspire other researchers to further develop an ISSHI. The architecture shall serve as a starting point for future research directions.

The presented requirements for the ISSHI are not free from limitations: The second interview series for evaluating and refining the requirements was relatively small compared to the first interview series. The refinements and additional requirements would likely be different. However, we selected a diverse sample of all layers of IT support and stopped the interview series because of conceptual saturation. Nevertheless, we plan to evaluate the architecture and first design principles with a bigger sample and will also concentrate on further evaluating/ refining the requirements. Second, the users of the IT support were mainly not considered. Following, asking the users about their perspectives could have led to different requirements. However, for the maximum impact regarding the IT support staff, we focused on support-agents and IT support managers.

8. Conclusion

With an ever-increasing number of IT systems, companies and their employees have to deal with, high-quality IT support that becomes a competitive factor for companies. The corresponding IT support organizations are faced with an increasing load of support requests. Due to the constant influx of new technologies, the types of support tickets are also becoming increasingly heterogeneous. Downtime of IT infrastructure and services poses a serious problem for operations of companies since downtime can quickly lead to high costs due to losing the ability to act/ work. HI is supposed to help in this context and to increase knowledge sharing within the organization while relieving the whole IT support staff from monotonous work, with high customer satisfaction. To derive the requirements and to design an architecture this paper followed a DSR approach. Literature in the intersection of AI, knowledge management, information retrieval, ITSM, and statistics, is combined with practice based on 29 in-depth interviews with stakeholders of the IT support from three companies.

Overall, the results of this paper contribute by a set of rigorously derived requirements as well as deep insights about challenges IT support stakeholders are facing. Furthermore, an architecture of an ISSHI to fulfill the broad requirements of the IT support staff was presented. This architecture reveals multiple fields for improvement and future research from a practical and scientific viewpoint. Finally, this paper shows how to combine artificial and human intelligence in IT support to (1) use the strengths of each actor, (2) teach the AI, and (3) increase IT support quality.

In conclusion ISSHI systems are service systems that integrate different resources [29, 30], which calls for the design of future work systems [31] and opens new avenues for modeling such complex resource relationships [32, 33].

References

[1] Hevner, A.R., "A three cycle view of design science research", Scandinavian journal of information systems, 19(2), 2007, pp. 87–92.

[2] Roshdi, A. and A. Roohparvar, "Review: Information retrieval techniques and applications", International Journal of Computer Networks and Communications Security, 3(9), 2015, pp. 373–377.

[3] Zhou, W., W. Xue, R. Baral, Q. Wang, C. Zeng, T. Li, J. Xu, Z. Liu, L. Shwartz, and G. Ya. Grabarnik, "STAR: A system for ticket analysis and resolution", Proceedings of the 23rd ACM, 2017, pp. 2181–2190.

[4] Shao, Q., Y. Chen, S. Tao, X. Yan, and N. Anerousis, "EasyTicket", Proceedings of the VLDB Endowment, 1(2), 2008, pp. 1436–1439.

[5] Xu, J., R. He, W. Zhou, and T. Li, "Trouble Ticket Routing Models and Their Applications", IEEE Transactions on Network and Service Management, 15(2), 2018, pp. 530–543.

[6] Kamar, E., "Hybrid workplaces of the future", XRDS: Crossroads, The ACM Magazine for Students, 23(2), 2016, pp. 22–25.

[7] Dellermann, D., P. Ebel, M. Söllner, and J.M. Leimeister, "Hybrid Intelligence", Business & Information Systems Engineering, 61(5), 2019, pp. 637–643.

[8] Boyce, C. and P. Neale, "Conducting in-depth interviews: A guide for designing and conducting in-depth interviews for evaluation input", 2006.

[9] Petter, S., W. DeLone, and E.R. McLean, "Information Systems Success: The Quest for the Independent Variables", Journal of Management Information Systems, 29(4), 2013, pp. 7–62.

[10] Nunamaker, J.F., R.O. Briggs, D.C. Derrick, and G. Schwabe, "The Last Research Mile: Achieving Both Rigor and Relevance in Information Systems Research", Journal of Management Information Systems, 32(3), 2015, pp. 10–47.

[11] Gregor, S. and A.R. Hevner, "Positioning and Presenting Design Science Research for Maximum Impact", Mis Quarterly, 37(2), 2013, pp. 337–355.

[12] Böhmann, T., J.M. Leimeister, and K. Möslein, "Service Systems Engineering", Business & Information Systems Engineering, 6(2), 2014, pp. 73–79.

[13] Wulf, J. and I. Blohm, "Service Innovation through Application Programming Interfaces - Towards a Typology of Service Designs", ICIS 2017 Proceedings, 2017.

[14] Eisenmann, T., G. Parker, and M. van Alstyne, "Platform envelopment", Strategic Management Journal, 32(12), 2011, pp. 1270–1285.

[15] Tiwana, A. and B. Konsynski, "Complementarities Between Organizational IT Architecture and Governance Structure", Information Systems Research, 21(2), 2010, pp. 288–304.

[16] Kubiak, P. and S. Rass, "An Overview of Data-Driven Techniques for IT-Service-Management", IEEE Access, 6, 2018, pp. 63664–63688.

[17] Pultorak, D., C. Henry, and P. Leenards, Microsoft Operations Framework 4.0 : a pocket guide, Van Haren, 2008.

[18] Potgieter, B.C., J.H. Botha, and C. Lew, "Evidence that use of the ITIL framework is effective", Conference of the national advisory committee on computing qualifications, 2005, pp. 160–167.

[19] Hochstein, A., G. Tamm, and W. Brenner, "Service oriented IT management: Benefit, Cost and Success Factors", European Conference on Information Systems, 2005.

[20] Alonso-Abad, J.M., C. López-Nozal, J.M. Maudes-Raedo, and R. Marticorena-Sánchez, "Label prediction on issue tracking systems using text mining", Progress in Artificial Intelligence, 8(3), 2019, pp. 325–342.

[21] Liu, J., Y. Cao, C.-Y. Lin, Y. Huang, and M. Zhou, "Low-Quality Product Review Detection in Opinion Summarization", Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2007, pp. 334–342.

[22] Rhyn, M. and I. Blohm, "A Machine Learning Approach for Classifying Textual Data in Crowdsourcing", Wirtschaftsinformatik 2017 Proceedings, 2017.

[23] Otterbacher, J., "'Helpfulness' in online communities", in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. 2009.

[24] Følstad, A. and P.B. Brandtzæg, "Chatbots and the new world of HCI", Interactions, 24(4), 2017, pp. 38–42.

[25] Xu, A., Z. Liu, Y. Guo, V. Sinha, and R. Akkiraju, "A New Chatbot for Customer Service on Social Media", CHI Conference on Human Factors in Computing Systems, 2017, pp. 3506–3510.

[26] Brandtzaeg, P.B. and A. Følstad, "Why People Use Chatbots", International Conference on Internet Science, 2017, pp. 377–392.

[27] Gnewuch, U., S. Morana, and A. Maedche, "Towards Designing Cooperative and Social Conversational Agents for Customer Service", International Conference on Information Systems, 2017.

[28] Sonnenberg, C. and J. Vom Brocke, "Evaluations in the Science of the Artificial – Reconsidering the Build-Evaluate Pattern in Design Science Research", International Conference on Design Science Research in Information Systems, 2012, pp. 381–397.

[29] Kleinschmidt, S., C. Peters, and J.M. Leimeister, "How to scale up contact-intensive services: ICT-enabled service innovation", Journal of Service Management, 31(4), 2019, pp.793–814.

[30] Li, M. M., C. Peters, and J.M. Leimeister, "Designing a peer-based support system to support shakedown", International Conference on Information Systems, 2017.

[31] Peters, C., "Designing Work and Service Systems", Habilitation Thesis, University of St. Gallen, 2021.

[32] Li, M. M., and C. Peters, "Reconceptualizing Service Systems – Introducing Service System Graphs", International Conference on Information Systems, 2018.

[33] Li, M. M., and C. Peters, "From Service Systems Engineering To Service Innovation – A Modeling Approach", Proceedings of the 27th European Conference on Information Systems, 2019.