2021 IEEE 25th International Enterprise Distributed Object Computing Workshop (EDOCW) | 978-1-6654-4488-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/EDOCW52865.2021.00056

Digital IT Consulting Service Provisioning – A practice-driven platform architecture proposal

Meikel Bode
Services & Cyber Security Group
University of Twente
Enschede, The Netherlands
m.bode@utwente.nl

Maya Deneva
Services & Cyber Security Group
University of Twente
Enschede, The Netherlands
m.daneva@utwente.nl

Marten J. van Sinderen
Services & Cyber Security Group
University of Twente
Enschede, The Netherlands
m.j.vansinderen@utwente.nl

***Abstract*—Although the digitization of their clients' business processes is one of the main areas of activity for IT consulting companies, the digitization of their own services is still largely absent. Even though we witness extensive use of digital tools, such as digital presentations or video meetings, in the daily work of IT consultants, the business is still very human-centric and dependent on their face-to-face interactions. This is not inherently a problem, yet it poses major challenges to the domain for which few solutions exist to date - the ubiquitous provisioning of digital IT consulting services with the help of digital consulting platforms. To mitigate this shortcoming, in this paper we propose an architecture design for a digital IT consulting platform. Using the design science methodology and based on high-level requirements elicited in an exploratory focus group study with 16 IT consulting practitioners, we derive data requirements, functional requirements, and quality requirements. Based on these requirements, we then propose necessary architectural components. The technology choices in the instantiation of our proposed architecture are illustrated in a realistic context. Our plan for follow-up evaluation is presented as well, along with our reflection on limitations and implications. With this work we make two main contributions: (a) catalog of derived requirements for a digital IT Consulting platform, and (b) a component-level architecture proposal of how a digital IT consulting platform can be designed.**

***Index Terms*—digital transformation, IT consulting services, IT service provisioning, digital consulting platform, architecture model, design science, requirements engineering.**

I. Introduction

IT consulting (ITC) is the business that helps companies digitally transform their businesses and processes. While much has been published on the digitalization going on in the other service business sectors, e.g. banking or insurance [4, 6, 8, 15, 17] that lead to terms such as FinTech [3], only little research has so far been done to understand how the digitalization of IT consulting itself is conducted and what - platforms are required for this to happen [5, 12, 16]. Digitalization of ITC, as we understand it, is not the simple shift from face-to-face interaction of client and consultant working together in the same room at the same time, to a video-conference-based setting and collaboration through a digital workspace. It is in fact the transformation of the ITC services to their digital "successors" whose provisioning as well as the orchestration of all involved actors, either human or technical systems, happens by means of digital ITC platforms. This paper addresses the need for more research on digital consulting platforms [11]. We felt motivated to approach the design of ITC platforms from an architectural perspective. If organizations would have proposals for architectural components, they would have a ready-to use starting point to consider when initiating a digital transformation journey. To this end, we set out to answer the following research question (RQ):

RQ1 *How can an architecture of a digital ITC platform for the provision of digital ITC services be designed based on ITC practice-informed requirements?*

The focus of this work is on deriving a possible architecture from previously collected requirements that might be expandable into a reference architecture in the context of future scientific work as well as ITC practice-oriented work. To achieve this, we start on the premise that a digital ITC service platform exists for a reason: the computer-aided provisioning of digital ITC services. We consider this realization critical to our work. It means that to be able to design a possible digital ITC service provisioning platform architecture, one must first consider digital ITC services. While we acknowledge that digital ITC services and digital ITC-service-provisioning platforms are interdependent, in the research scope of this paper we will not treat digital ITC services in detail. Instead, our focus will be on digital ITC platforms. To this end, we assume that digital ITC services descriptions (DSDs) are available and formulated in a computer interpretable notation. To answer our RQ, methodologically, we are guided by the approach of design science [7, 13, 18]; in line with this, our intended aim is the design of a practice-driven artifacts, namely an architecture model.

The rest of the paper is organized as follows. In section II we describe the used research method. In section III we present our architecture proposal. We continue in section IV with an illustration of one possible way to instantiate our architecture model. In section V we present our scenario and the intended research methods for demonstration and evaluation of our proposed architecture. Finally, we conclude and discuss limitations and implications of this work in section VI.

II. Research Methods

As already indicated, our research process has been inspired by methodological sources on design science [7, 13, 18].

978-1-6654-4488-0/21/$31.00 ©2021 IEEE
DOI 10.1109/EDOCW52865.2021.00056

Following Peffers et al. a design science research cycle consists out of six phases: (1) problem identification, (2) definition of objectives, (3a) design & (3b) development, (4) demonstration, (5) evaluation, and (6) communication [13]. In this paper, we report the proposal for architecture (our artifact) grounded in our problem analysis, along with an illustration of this architecture's instantiation. Because of the page limit, we report our plan for the systematic evaluation of this architecture in follow-up case studies.

A. Problem Identification

ITC is using digital technologies and current forms of communication extensively already. But when it comes to the ITC services and processes, the digitalization becomes scarce, and the domain is still person-centered beyond measure. The awareness that the central consulting processes and services in the ITC domain have not yet been digitalized to any great extent and that this can lead to various drawbacks especially in the current pandemic, is something the first author has been able to develop over several years in his role of an IT consultant. In fact, this observation triggered the initiation of this research. We consider the motivation for this work to be problem-centric [13] as it is linked to the desire to verify this personal perception and experience. Following [13], we went on taking two systematic steps to understand the state-of-the art regarding the problem. First, we conducted a systematic literature review [1] on the state-of-the-art research related to the topic digitalization of ITC, which found that little has been published so far on the design and empirical evaluation of architecture and platform related artifacts. Second, we conducted an exploratory focus group (FG) with 16 ITC practitioners to understand what IT-consultants think of the requirements for digital IT services. Our FG results are presented in [2]. Therein, we reported 79 requirements for digital ITC services and digital ITC service platforms. The results from the conducted FGs strengthened our motivation and deepened our understanding of the problem. The FG participants all agreed that platform development is long overdue and that it is urgent now that requirements for these platforms are well-understood and documented. If a common understanding of the requirements exists – and proposals for architecture components are made, the ITC sector can collectively make progress towards the digitalization of its own service delivery processes, and in turn towards reducing cost and increasing delivery speed. This paper reports on our follow-up action in response to the problem of lacking platforms.

B. Definition of Objectives and Solutions

The objective of this work is to propose a solution to the expectations and requirements from ITC practitioners [2] by mapping them into a suggested digital ITC platform architecture model. The proposed architectural model can then later be instantiated for evaluation [18] in a real-world context (section V) in configuration levels of different complexity.

C. Requirements Analysis

The basis for the design phase of our proposed artifact (the architecture) are the 79 requirements collected in [2]. To be able to consider or reject candidate requirements for inclusion in our proposed architecture and to be able to derive architectural decisions appropriately, our first step is the categorization of the requirements into the classes as proposed by Lauesen: data requirements (DR), functional requirements (FR), and quality requirements (QR) [9]. In each class, we started with the raw source requirements elicited from the FG practitioners [2] and based on these we formulated so-called *Derived Architecture Requirements* (DARs). These are presented in Table I, Table II, and Table III. These DARs are our original contribution which we use as the foundation for the architecture models that will be presented later in the paper. To assure the traceability between the elicited "source requirements" and the DARs, in each row of Table I, Table II, and Table III, we first state the "derived architectural requirement" (DAR) that we developed from one or more source requirements. Right after each DAR, we state in italic the FG results, that are our "source requirements" and contain the original requirement statement used in [2]. These DARs are then later referred to in our architectural model (section III). The formulated DARs describe the requirements identified in [2] to varying degrees of concreteness, so that they can be used as a basis for defining architecture components. In the context of a real instantiation, it is to be expected that they can and will be deviated from.

1) Data Requirements: DR deal with needs related to data persistence, data formats, data model, data input and output, and storing system as well as communication states [9]. The DRs are closely related to the FRs. We identified 9 DRs (Table I) from the original source requirements but we do not claim completeness. Rather, we expect to identify additional during an instantiation. We think that this choice is justifiable as different ITC organizations might choose different supporting technologies. As we will see in section III, while instantiation an architecture model, one faces some specific decisions, which might lead to specific DRs, respectively.

2) Functional Requirements: FR describe the functionality of a system related to its abilities to process, persist, exchange, and manipulate data [9]. Functional requirements often are implemented through interfaces, whereas the user interface is of high importance [9]. We identified 18 FRs (Table II).

3) Quality Requirements: QRs, also called non-functional requirements, are closely related to the FRs. They describe how well the FRs (should) perform their intended functionality [9]. We identified 16 QRs (Table III).

III. Design of our proposed Architectural Model

This section presents an initial proposal for an architectural model of a possible digital ITC platform based on our collected requirements. The architecture includes a set of decoupled and potentially distributed components that interact with each other through interfaces that provide access to component functionalities and data entities. Our proposed components

TABLE I: Data Requirements

ID	**Derived Architectural Requirement (DAR)** *Source Requirements [2]*
DR1	Store standardized ITC DSDs modules and packages in a repository. *ITC Service structure (modularization, packaging, standardization)*
DR2	Store DSDs client specific requirements for digital ITC service and related customizing data. *Requirements management*
DR3	Store ITC service provisioning relevant measuring data. *Measurability*
DR4	Store different classes of actors and roles (e.g., companies, persons, technical systems, users). *Actors*
DR5	Store sensitive data encrypted. *Security; Protection from theft; Data protection*
DR6	Store technical, methodical, and experiential knowledge related to ITC and digital ITC service provisioning. *Access to collective knowledge and experience; Collect & provide knowledge between consultants*
DR7	Store documentation and guidance relevant data. *Guide the client; Guide the consultant*
DR8	Store digital ITC service provisioning relevant key figures for later processing as such as billing, service optimization, reporting. *Guide the client; Guide the consultant*
DR9	Whilst the digital ITC platform components communicate through internal interfaces, the platform must provide an API for external communication. External communication partners can be other systems of the IT consultancy, but also third-party systems, e.g., of clients or partners. The API provider should provide secure and restricted access to relevant entities of the platform, e.g., actors, roles, DSD, instances of running digital ITC services, collected data on the service provisioning process, messaging etc. The access to entities and functionalities must be protected by an IAM. The data formats supported by the API can vary from XML, JSON, YAML. The preferred communication protocol should be HTTP-based like REST. But also, proprietary protocols and formats, such as SAP Remote Function Call (RFC) should be possible. The API layer should translate from internal to external protocols and vice versa. *Application Programming Interface (API)*

are described in Table IV. For each component, we put the original requirement referenced in italics below the DAR. We note that in Table IV, we make a distinction between *relevant* and *optional* components. We think that this is important for later use of our model in a process of instantiating the architecture in specific contexts. The decision what is required and what is not has to be taken in a concrete instantiation context. Additionally, we mark components as *in scope* for our subsequent research activities related to validation. Our proposal for architecture is depicted in Figure 1. We group

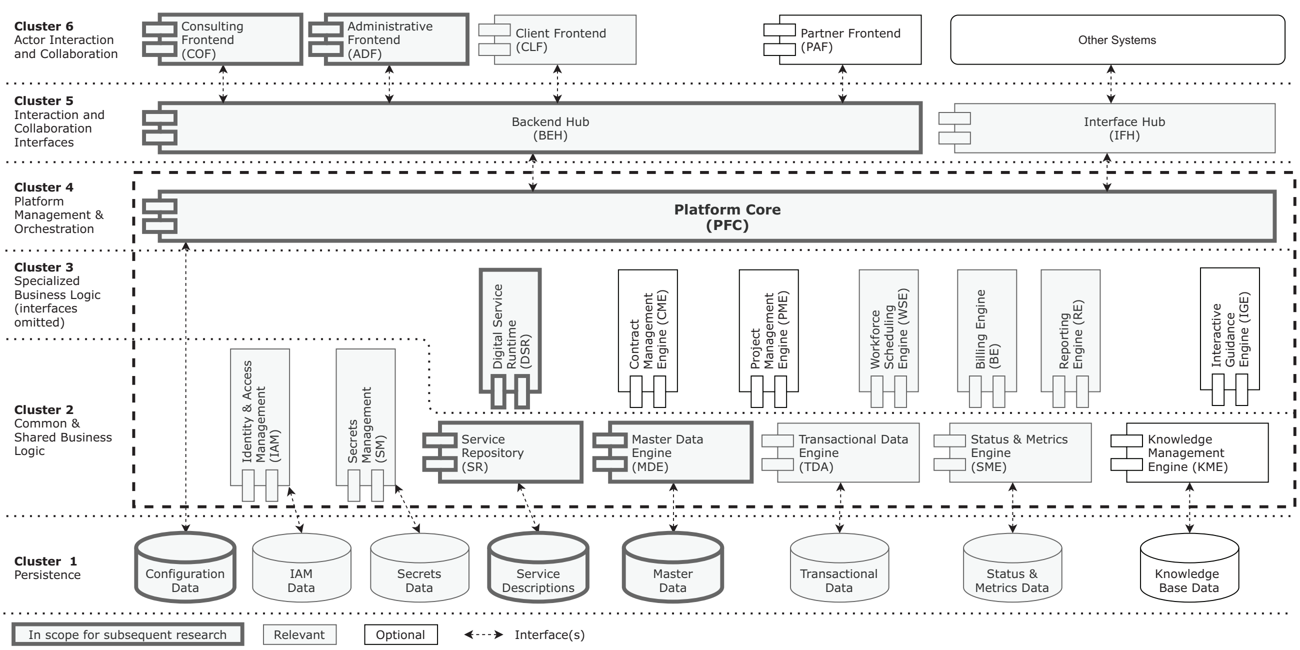

Fig. 1: The proposed Architecture for a digital IT Consulting Platform

the architecture into six component clusters. We describe these below. We note that in the description, we use the acronyms that are given in the first column of Table IV.

Cluster 1 contains the data storage components. Cluster 2 contains components such as the SR or the MDE that directly interact with the data storages and provide access to higher level components via interfaces. Cluster 3 contains specialized components more related to business logic such as the DSR or the WSE. These types of components consume functionality provided by cluster 2 components. For simplification reasons we omit the interfaces between the platform core (dashed rectangle) and other central components. All central components provide interfaces to their contained business logic to the other components. Cluster 4 contains the PFC component that is responsible for the registration, the monitoring, and the orchestration of all other platform components. Cluster 5 contains two components providing interfaces related to actor interaction and collaboration. The BEH provides interfaces to the various frontends, whereas the IFH provides interfaces to other infrastructure such as internal systems of the IT consultancy, to partners, and to clients. Cluster 6 contains the components used by actors to interact with the platform. These might be frontends but also other systems that consume interfaces provided by cluster 5 components.

TABLE II: Functional Requirements

ID	Derived Architectural Requirement (DAR) *Source Requirements [2]*
FR1	Support the maintenance (creation, change, deletion) of digital service descriptions (DSDs). *ITC Service structure*
FR2	Inform interested actors continuously regarding status changes of running ITC service instances throughout the provisioning process. This can be supported by specific UI functionality and by using other communication channels like email, active chat, or other messenger services. *Transparency of service provisioning*
FR3	Monitor running ITC service instances and remind incorporated human actors to fulfill their assigned tasks by email or messenger. Inform also coordinating actors to potentially reassign tasks to other consultants or by deciding automatically on alternative actors. *Faster Service Provisioning*
FR4	Implement the functionality to collect data on all phases of the service delivery process to be able to tell interested actors key performance indicators. *Measurability*
FR5	Implement all features and functionality of the digital ITC platform with security in mind in the sense of "security by design". *Security; Secure*
FR6	DSDs and the digital ITC platform shall support the automated testing and documentation of the result produced during the services provisioning process. This might be reached by defining certain conditions that must be meat and could be automatically evaluated. E.g., a certain configuration set was implemented in a customer system which could be checked by calling an API in the customer system. The achieved consulting results as well as the service provisioning process should be visualized. *Automated testing of consulting results; Automated documentation; Result visualization*
FR7	In the ITC domain different consultancies jointly provide digital ITC services to one customer, for example in a project setting. The digital ITC platform must be able to interact with digital ITC platforms / systems of other consultancies and share services descriptions as well as status updates. *Cooperative service delivery in partner network; Market place*
FR8	Implement interview functionality aiming incorporated actors for the sales and marketing, as well as services provisioning process to collect experience and learnings continuously and in a structured way. Implement a knowledge base functionality for consultants to access knowledge and learnings achieved in the past. *Access to collective knowledge and experience*
FR9	The digital ITC platform supports the definition of teams (e.g., service specific or project specific) and the orchestration of those teams related to the digital services provisioning process. This includes knowledge about team members and their skills and skill levels or their experience with digital services provisioning. The platform should be able to incorporate (dispose, assign, replace, etc.) team members into the services provisioning based on the requirements of digital ITC services running in a certain context. The platform therefore must know the schedule and availability of each team member at any time. The platform must collect information on assigned tasks frequently from all members to compute the degree of completion of any running service instance at any time. *Orchestrate international virtual teams*
FR10	The digital ITC platform should support the client with functionality to simulate the effect digital ITC services would have when applied to their environment, e.g., when implementing a chart of accounts in an SAP system beforehand. This requirement holds also for consultants who need to verify that a certain digital ITC services is valuable for a certain client. Additionally, support the maintenance of customer systems and their required technical configurations, e.g., hostname, user credentials etc. These systems might be used during the service provisioning process to automatically retrieve information, apply customizing or configuration data or trigger activities on those systems. *Enable clients to simulate & verify requirements*
FR11	Implement a functionality that helps to understand how mature a potential client is regarding its digital readiness. It is important to understand what level of digital maturity potential clients have already reached to decide which service activities can be externalized to them and which actions they can perform on their own or how much active assistance must be provided by either the digital ITC platform or consultants. The grading should be repeated on regular basis to allow clients to grade up or down. Additionally, actively open the communication channel to clients on a regular basis to make sure the clients are kept informed and the service provisioning process is always transparent to them. *Grade the client; Partially takeover of mental transfer; Externalize service activities to client; Continuously reconnect with client*

Continue on the next page

TABLE II: Functional Requirements (continued)

ID	**Derived Architectural Requirement (DAR)** *Source Requirements [2]*
FR12	During the service provisioning the running service instances must always be accessible by consultants on duty in case help is required by clients. The functionality should support an in-personate functionality to see the service instance from a different actor's (e.g., the client) perspective. *Consultant step-in always possible*
FR13	Implement functionality that helps incorporated actors to easily transit through the services provisioning process. This functionality is meant to be a context-sensitive help / guidance system that acts in the background and actively or on demand steps into the foreground to provide help and guidance. *Guide the client; Guide the consultant*
FR14	Design the architectural components in a way that the collection of runtime-related data is supported by default (software and resource measurement and metrics) that can be used for later reporting and billing. *Collect billing relevant data*
FR15	Support the download of content by interested actors. *Download of examples*
FR16	Design and implement the digital ITC platform so that it can be handled/used like a web-shop and supports well-established concepts such as: product/service catalog, a product/service configurator, pricing engine, user authentication, main-/sub-accounts for clients, payment. The system should make it easy to find the required digital ITC services by providing a comprehensive search system for the service catalogs. *Web-shop; Easy to find services*
FR17	The user experience must be supported by a modern and intuitive and responsive user interface. Use modern web-based standards to build the different user interfaces for the different roles working with the digital ITC platform tool. The overall system architecture should follow modern, barrier-free architectural design principles and should be decomposed into modules. The user interfaces should be realized as one decoupled module or could also be realized several times for different audience/purpose/platforms, e.g., like web-based mobile or personal computer access through browsers, specialized mobile applications or as a dedicated software component installed on users' desktop PCs. *Modern and responsive user interface; Accessibility; Device independent availability; Barrier-free*
FR18	It should be possible to describe and implement new DSDs within the digital ITC platform using the no-code/low-code principle. The idea behind this requirement is that even normal consultants should be able to design new DSDs without attending intensive trainings beforehand. *No-code/low-code*

IV. Illustration of the Architecture Instantiation

This section illustrates possible ways to implement our proposed architecture. It relates to the development phase of the design science process. The following subsections discuss our first concrete ideas of, how the defined architectural components could be implemented. In general, we plan to utilize existing and proven tools, common programming languages, and available frameworks wherever possible. Although the following sections provide general considerations of what a full implementation of a digital ITC platform might look like and how certain components or clusters shall be implemented, the early instances of the platform will be less complex prototypes consisting out of the components marked as *in scope*: *service repository*, *master data engine*, *digtal service runtime* as well as required parts of the *platform core*, *backend hub*, and *front ends* (Figure 1). We think that this will be sufficient for the very first instantiation as our focus lays on demonstration and subsequent evaluation. Later iterations will add more components or change the current ideas and made considerations might have to be adopted accordingly. In what follows we summarize our considerations regarding (1) platform runtime and orchestration, (2) storage and persistence, (3) backends, (4) frontends, (5) IAM, (6) status and metrics, and (7) interfacing.

A. Platform Runtime and Orchestration Considerations

For the very first instantiation, we suggest selected components to be packaged using a common container technology such as Docker (https://www.docker.com). This will help to speed up the development and testing cycles because all relevant configurations get packaged into the container and can be automatically renewed and rebuild whenever necessary. The architecture will be orchestrated by an engine such as Kubernetes (https://kubernetes.io). Kubernetes supports the setup of clusters of distributed nodes that act as one logical unit. An alternative orchestrations engine could be Docker Swarm (https://docs.docker.com/engine/swarm) but due to its declining relevance it is not a valid option. Both alternatives would help to define scaling options for deployed architecture components as required, e.g., if a certain threshold of maximum parallel running digital ITC services would be reached, additional components could be automatically provisioned by the orchestration engine.

B. Storage and Persistence Considerations

We choose PostgreSQL (https://www.postgresql.org) as the database for persistence requirements of our components. PostgreSQL is an open-source relational database management system (RDBMS) that has proven its reliability and performance since many years. Alternatively, we could use MySQL (https://www.mysql.com), or any other existing free or commercial RDBMS to reach our goal. Since the main author as has good experience with PostgreSQL, we opt for it.

C. Backend Component Development Considerations

Many architecture components must be explicitly designed and then implemented. We choose Java as the programming language because the main author is very experienced in this language. Java is a common object-oriented language with a lively community and a rich set of available frameworks and tools. The backend components in the clusters 2, 3, 4, and 5 will be implemented using the Spring Boot (https://spring.io/projects/spring-boot) framework. Spring Boot offers a lively community and allows the fast setup and development of web-applications which can easily expose a RESTful API.

D. Frontend Component Development Considerations

Our frontends will be build using Angular 12 (https://angular.io). Angular offers the model-view-controller pattern. Angular apps are programmed using TypeScript and are later compiled to Java Script. This makes it a superior choice over manually programmed Java Script since it reduces errors due to its strong typing. Angular apps are loaded from a web location into the web browser's memory and are being executed on the client's devices, such as mobile phones or desktop computers. The frontends communicate with the digital ITC platform via backends through RESTful APIs provided by the cluster 5 components. Alternatively, our frontends could be based on server-side rendered frontend technologies, such as Java Server Faces or Spring Thyme Leafe.

E. Identity and Access Management Considerations

For the IAM component we choose Keycloak (https://www.keycloak.org) as an existing and proven tool. Keycloak offers user, role and client management functionalities paired with latest OAuth2.0 and OpenID Connect authentication schemes and build-in multi-factor-authentication based on one-time-pad. Through different authentication flows various types of clients, such as web frontends or technical services can be integrated into the overall architecture using the same security standards.

TABLE III: Quality Requirements

ID	**Derived Architectural Requirement (DAR)** *Source Requirements [2]*
QR1	Support the scaling of the digital service provisioning process by means of performant, stable and parallel execution, and the automated upscaling of compute and storage resources. *Scalability; Performance; Stability*
QR2	Execute the digital ITC service provisioning process in a predictable manner and reach higher service quality by learning from previous process executions. *Higher Service Quality*
QR3	Provide insights into the digital ITC service provisioning process continuously to interested actors. The digital ITC platform will provide new digital ITC services and many activities will be executed in the background so that incorporated actors must trust the platform to deliver high-quality results. The platform must therefore be stable, reliable, and responsive during user interaction. *Transparency of service provisioning; Trustworthiness*
QR4	The digital ITC platform must be self-explaining to its users. Keep simplicity in mind related to every architectural decision. Define and design all functionality in the same way and reuse design elements throughout the platform. *Self-explaining*
QR5	All incorporated actors need to be able to access relevant services through communication or interaction channels. This might be an UI for human actors that helps browsing a catalog of DSDs/inspecting running digital ITC services or an API that helps technical actors/third party systems to discover available services and to interact with running digital ITC service instances. *Accessibility of the services*
QR6	New actors must be on-boarded and get used to the digital ITC platform fast. Additionally, the ITC platform must be easily to be ramped-up and to be adopted to new customers and their requirements on the digital conduction of ITC projects. This could be reached by using established standards for example in the used technologies, communication and data formats, and programming languages or by sticking to best-practices. *Fast on-boarding & ramp-up; Consultant introduces client to platform & services*
QR7	The scope and value to be expected by clients must always be clear to them. Therefore, it is necessary to communicate the value proposition to the client continuously and at an early stage. DSD must therefore be designed with their value proposition in mind, and this must be continued throughout the service provisioning process e.g., by transparently communicating the already reached value grade. *Value proposition must always be clear to client*
QR8	The digital ITC platform must be easy to use and to consume by clients which have the required digital maturity. This means that the necessary understanding on client-side must be established how to interact with digital ITC services as well as digital ITC platforms. *Easy to use; Easy to consume;*
QR9	The digital ITC platform should help consultants with their repeating every-day problems and actively support the consultants in their work. Consultants will use the system regularly and therefore it should be easy to use and should motivate to use it in the sense that it makes the consultants' job easier. The system therefore must not be obtrusive, e.g., by not enforcing the communication with incorporated actors, but be available as a guide whenever necessary. *Help with repeating problems; Supportive tool for consultant; Must make fun; Must not be obtrusive*
QR10	Incorporated actors, e.g., consultants and clients should have access to the knowledge base and experience made by consultants and clients of the company in the past. This could be supported by actively interviewing incorporated actors before, during, or after the sales and marketing, as well as the service provisioning process regarding their expectations and achieved learnings. *Access to collective knowledge and experience*
QR11	The digital ITC platform should handle activities within it-self and reduce media breaks, e.g., the switch between different tools to a minimum. *No media breaks*

Continue on the next page

TABLE III: Quality Requirements (continued)

ID	Derived Architectural Requirement (DAR) *Source Requirements [2]*
QR12	A digital ITC platform must lead to a faster service provisioning process compared to classical ITC service provisioning and should support the overall consulting process end-to-end. It also should reduce the number of different tools to be used by consultants and lead to a tool harmonization. This aim can be reached by actively checking the current phase of running digital ITC service instances and their provisioning process and by reminding human actors (consultants as well as client personnel) continuously to fulfill assigned activities. *Faster Service Provisioning; Higher Effectiveness; Manage the end-to-end consulting process; Harmonization and reduction of tools*
QR13	Increased virtualization of ITC services leads to lower service costs and higher margins. To reach this a digital ITC platform must be designed so, that digital ITC services can be delivered at lower prices. E.g., by the support of automated services provisioning and the partial replacement of human with digital actors. *Low Costs; Higher Margin; Higher efficiency*
QR14	Use the digital ITC platform and affiliated digital ITC services to establish a new and trusted consulting brand and use it as a successful and unique selling proposition that disrupts classical non-digital equivalents. The better the experience is and quality of digitally provided ITC services is the higher is the chance that clients request additional digital ITC services. To develop a trustful relationship with the clients they should be integrated into the service provisioning process at an early stage. This might also be supported when a personal interaction with a consultant takes place at the beginning of a digital service provisioning. *Create Consulting Brand; Must have stickiness to clients; Success higher when trust exists; Digital offerings disrupt non-digital equivalents; Integration of client early; Personal interaction at the beginning*
QR15	By using a digital ITC platform and corresponding services, better understand what customers want and need, and adopt the consultancies ITC services portfolio accordingly and fast. *Demand-driven services*
QR16	Digital ITC platforms are seen as the enabler for digital consulting according to [2]. That means that consulting companies shall introduce a digital ITC platform early to enable themselves to experiment with digital ITC service provisioning and gain experience. *Enabler for digital consulting; New firms react better then incumbents*

F. Status and Metrics Collection Considerations

The continuous monitoring to ensure the flawless operation of our digital ITC platforms and its components as well as the running digital ITC service instances, is crucial for the continuous success and acceptance of a digitalized ITC business. For this purpose, we will use the open-source monitoring solution Prometheus (https://prometheus.io) to support this requirement. Prometheus is a server component that can receive or pull metrics from connected components. All collected data are stored in a time series database for later analysis and consumption e.g., by the BE. Through additional frameworks such as Micrometer, Prometheus monitoring can be integrated in Spring Boot applications. Plus, the central orchestration engine Kubernetes, the used persistence technology, as well as other components can be connected using special exporters to Prometheus. Thus, Prometheus represents an overarching and integrated solution.

G. Interface Hub Considerations

The digital ITC platform must ensure the smooth technical integration of heterogeneous external system landscapes with software solutions from different manufacturers and different releases. In addition, it must support different security standards, communication protocols, data formats and encoding schemes. In line with this, as an elementary building block of our Interface Hub component, we select Apache Camel (https://camel.apache.org). It is an open-source communication platform that already supports more than 50 data formats and can also be extended. The current development is increasingly moving towards the provision and use of stream-based communication. One of the most widely used solutions is Apache Kafka (https://kafka.apache.org). Kafka is open source distributed platform. As part of a future design iteration, it may seem reasonable to integrate a streaming solution into the Interface Hub.

V. Demonstraction Scenario

As the scope of this paper is the derivation of the DARs and the proposal of architecture, we think it is important to supplement them with a plan for demonstrating a digital platform prototype implementing the proposed architecture. This plan forms our immediate future research activity. The next subsection describes the real-world context where our demonstration scenario will be executed. This is followed by our evaluation plan.

A. Demonstraction Scenario

1) Context: SAP Financial Consulting: Following the design science process [13, 18] the demonstration of the prototypical digital ITC platform (artifact) to be developed will be performed within a defined context and component scope (Figure 1). We choose the area of SAP Financial Consulting for this purpose, since this consulting field is characterized by an already high level of standardization (e.g., accounts payable, accounts receivable), which can be justified by legal requirements. Furthermore, this consulting field makes sense since the first author is actively participating in projects in this area, and, in turn, has many points of contact in the context of his professional activity.

2) Usage of the digital ITC Platform Prototype: The prototype will be placed as one of the tools available to SAP Financials consultants. The platform will be introduced to the consultants, and they will use it to provide digital ITC services. Iteratively, the consultants will use the platform and provide services in test scenarios and learn how to interact with the platform and the service descriptions. The prototype will not

TABLE IV: Architectural Components

Component	Description *Referenced DARs*
Platform Core (PFC)	The platform core provides common functionality to the other architecture component such as central configurations for distributed components. All other components of the architecture must be registered with the PFC. The PFC manages the other components and monitors their health status. Hence, the PFC is a very complex component as it serves as the central integration point. *FR5, QR1, QR13, QR15*
Identity & Access Management (IAM)	The IAM provides identity and access verification and maintenance services to the other architecture components. *DR5, FR5*
Secrets Management (SM)	The SM provides functionality to encrypt, decrypt, and securely persist secrets, such as certificates, passwords, security tokens, or other sensitive information. *DR5, FR5*
Service Repository (SR)	The service repository contains all DSDs. It is responsible for the hole life cycle of a digital services description and manages their different versions. *DR1, FR1*
Digital Service Runtime (DST)	The digital service runtime implements functionalities to instantiate a DSD to a running digital ITC service instance. IT manages all relevant customer specifics related to a certain service instance. Additionally, the DST is responsible to take over the automated documentation of provisioned digital ITC service instances. The DST might be horizontally scaled to guaranty performant and responsive behavior at different locations world-wide. *DR2, FR3, FR4, FR5, FR6, FR9, FR14, QR1, QR2, QR12, QR13, QR15*
Workforce Scheduling Engine (WSE)	The WSE manages and orchestrates the available staff. The WSE assigns automatically or by manual decision staff to certain projects or independently running digital ITC service instances. The WSE has knowledge about consultant skills and availability. *DR4, DR5, FR3, FR4, FR5, FR7, FR9, FR14, QR12, QR13*
Contract Management Engine (CME)	The contract management engine holds contractual information relevant for the service delivery (due dates, rates and prices, dedicated personnel, service level agreements, or other resources). It is important to always know which deliverables (services or goods) are contractually owed to the customer. *DR5, FR14*
Project Management Engine (PME) *optional*	The PME holds information of historic, current, future internal, and external projects. The project master data can manually be entered or automatically derived from contractual data provided by the CME. *DR5, FR14*
Interactive Guidance Engine (IGE) *optional*	The IGE supports incorporated actors by providing context sensitive help or on dedicated request. The IGE uses information collected from either the SR, the KME, or other sources configured. In general, it should perceive the current situation an actor is in and (re)act appropriately. *DR6, DR7, FR4*
Status & Metrics Engine (SME)	The status and metrics engine implements a central sink for all status and metrics data collected from other architectural components and external through API connected technical actors. *DR3, DR5, DR8, FR2, FR4, FR5, FR14, QR2, QR3, QR15*
Billing Engine (BE)	The BE produces billing relevant invoicing items by consuming information provided by SME, the TDE, and CME. The BE ensures that all billable services, goods, and deliverables are identified and forwarded to an external invoicing system. *DR5, FR14, QR3*
Master Data Engine (MDE)	The master data engine is responsible for storing all relevant master data used by other architectural components. It can act as the leading system for master data but could also rely on other external systems. Contained master data type could be e.g., customers, actors, roles, article data, article, and service catalogs, etc. *DR5, QR1, QR3*

Continue on the next page

be interconnected with other systems and all activities will run inside the platform. It is intended that during the first demonstration phase also first actor orchestrations will take place. This means, that services with more than one actor (e.g., hand-over of service tasks from one consultant to another consultant) will be available and so become testable by the consultants and later clients.

B. Experimental Treatment Evaluation Plan

To evaluate our prototyped digital ITC platform artifact, we plan to apply a combination of the research methods of expert opinions and case-based experiments (as per Wieringa's suggestion [18]). Expert opinions form a practical method to collect judgment from domain experts, in our case SAP consultants, about our prototype placed in the problem context [14, 18]. As our prototype will be development in several iterations, we plan to collect expert opinions several times to continuously improve it. Case-based experiments can be used to validate the solution design in a simulated problem context [18]. By applying stimuli to the prototype and the subsequent observation of its responses, the researcher can understand the prototypes operational behavior and draw conclusions [18].

VI. Discussion and Conclusion

This work reported the results of a design science process aiming on the derivation of architecture requirements to design a system architecture proposal for a digital ITC platform. Based on 79 requirements collected in a former study [2] we developed 9 data requirements (Table I), 18 functional requirements (Table II), and 16 quality requirements (Table III),

TABLE IV: Architectural Components (continued)

Component	Description *Referenced DARs*
Transactional Data Engine (TDE)	The transactional data engine implements functionality related to fast changing data of running digital ITC service instances, consulting project data, order data coming from the web-shop engine and so forth. The implementation should consider concurrent data changes from multiple distributed DSTs related to the same service instance. *DR5, DR8, QR1, QR3*
Reporting Engine (RE)	This component prepares and renders reports for different stakeholders based on data collected by the SME online or in batch processing mode. The RE delivers reports to their respective receivers through different channels using the IFH. *DR5, FR2, FR6, FR11, QR3*
Knowledge Management Engine (KME) *optional*	The KME implements functionality to document and retrieve knowledge related to consulting activities. The KME data get consumed by the different frontend components. The KME supports the documentation based on common markup format like HTML or wiki-syntax. Additionally, files in common office formats (PDF, Word, Excel, etc.) can be stored in the KME and searched. *DR5, DR6, FR8*
Interface Hub (IFH)	The interface hub represents the central communication endpoint for external systems communicating with the platform. The IFH coordinates inbound and outbound traffic to and from trusted or untrusted parties. All communications are logged and can be monitored. *DR9, FR4, FR5, QR1, QR3, QR5*
Backend Hub (BEH)	The backend hub provides the internal business logic to the various frontends. *DR5, DR9, FR5, QR1, QR3*
Client Frontend (CLF) *optional*	The web-based client frontend that represents the main entry point for clients interacting with the digital ITC platform. *DR5, FR2, FR4, FR8, FR10, FR11, FR12, FR15, FR16, FR17, QR1, QR3, QR4, QR5, QR6, QR7, QR8, QR10, QR11, QR12, QR14, QR15*
Consulting Frontend (COF)	The web-based consulting frontend that represents the main entry point for consultancy personnel (e.g., consultants, accountants, dispatchers) interacting with the digital ITC platform. The COF also implements the functionality to design and maintain DSDs. *DR5, DR6, FR1, FR2, FR4, FR5, FR8, FR12, FR15, FR16, FR17, FR18, QR1, QR3, QR4, QR5, QR6, QR7, QR8, QR9, QR10, QR11, QR12, QR14, QR15*
Partner Frontend (PAF)	The web-based partner frontend that represents the main entry point for partner company personnel (e.g., consultants) interacting with the digital ITC platform. *DR5, DR6, FR2, FR4, FR5, FR7, FR8, FR12, FR15, FR16, FR17, QR1, QR3, QR4, QR5, QR6, QR8, QR9, QR10, QR11, QR12, QR14*
Administrative Frontend (ADF)	The web-based frontend that represents the entry point for administrative access to the digital ITC platform. *DR5, FR2, FR5, FR17, FR18, QR3, QR4, QR8, QR9, QR12*

which lead to 21 architectural components (Table IV). Furthermore, we assembled the components into an architecture proposal of a digital ITC platform (Figure 1). To illustrate the implementation of the architecture, we have provided examples of technology choices relevant for the components. As mentioned, our considerations for implementation options concerned a an in-scope subset of all possible components. Our immediate research plan is to implement a prototype platform in the context of an SAP consulting project in the financial accounting area to evaluate the usefulness and the utility of the proposed architecture in the field, from the perspective of IT consultants practitioners.

A. Limitations

Regarding our architectural proposal for a digital ITC platform, we acknowledge the following limitations [18] . First, would the DARs and the proposal of architecture be possible to implement to other ITC areas, beyond the one from which our source requirements [2] are elicited? We note that the underlying requirements, which we used as a source to derive our DARs (Table IV) have been identified in the context of focus groups conducted with practitioners, almost all of whom have a longstanding SAP consulting background [2]. In addition, the first author of this work also shares this background. Thus, we cannot exclude 100% the possibility that our platform design has a high specificity for the SAP consulting environment. However, we could possibly think that other IT consultants providing services concerning other IT solutions (like SAP services, but different) might provide similar requirements. For example, services concerning the implementation of Oracle, Microsoft Dynamics, Peoplesoft. Following Wieringa [18], we think that it might well be possible that similar processes of providing similar services would share similar requirements for digital platforms for IT service provisioning. Therefore, we tend to possibly believe that the DARs we derived might well be suitable for other consulting settings beyond those from which our original source information came from. Of course, more empirical evaluation research is needed to collect evidence about the transferability of our DARs and proposed architecture to other contexts. This forms a line for future research. Second, we see a possible limitation related to the availability of highly standardized ITC services [10]. Applicable ICT domains must be able to describe their consulting services at a high level of standardization so that they can be presented in a computer-interpretable format. This is the prerequisite for their provision via our platform. This limitation may be more applicable to established consultancies with established classical services and process models than to newcomer companies starting at a higher level of digital maturity [10]. The ideal IT consulting firm for our platform would therefore have a high level of digital maturity and have already structured, standardized, and modularized its ITC services to a large extent. In this case, the effort to digitize these services and provide them on our digital ITC platform would be expectedly lower. Nevertheless, we believe that our platform design can be generalized and be of practical use for other ITC consulting areas as well. This might also hold for research in digitization of related consulting fields as a valuable basis for our own research projects, either directly or adapted.

B. Implications for Research and Practice

This work has some implications for research and practice. First, with our catalog of DARs and our proposed ITC platform architecture we contribute actively to the body of scientific knowledge in the domains of Consulting Research as well as Information Systems Research. In fact, we responded to the call of Nissen et al. [11] for providing more scientifically grounded artifacts in digital transformation of consulting. Second, the DARs catalog can be used in future research on prototype development and evaluation of IT (consulting) platforms. To practitioners, our work has at least three implications. First, we consider the DARs catalogue as a candidate starting point for those practitioners engaged in the digitalization of the services in their own consulting organizations. Second, if consultants evaluate a candidate platform for use, they could possibly consider using the DARs catalogue as a basis for the evaluation of the extent to which the candidate platform meets the architectural requirements. Third, the proposed ITC platform architecture serves as an entry point for the design and implementation of ITC artifacts to support the domain regarding its digital transformation. Due to a potential transferability of our findings, they might also be relevant for other areas of the service sector such as banking or insurance.

References

[1] M. Bode, M. Daneva, and M. J. van Sinderen, "Characterizing the Digital Transformation of IT Consulting Services – Results from a Systematic Literature Review," Submitted to a Journal, Under Review, 2021.

[2] ——, "Requirements for Digital IT Consulting Services and their Provision through Digital Consulting Platforms - Results from a focus group study," in *23rd IEEE ICB*, Bolzano, 2021.

[3] K. H. Ching, A. P. Teoh, and A. Amran, "A Conceptual Model of Technology Factors to InsurTech Adoption by Value Chain Activities," in *2020 IEEE Conference on E-Learning, e-Management and e-Services (IC3e)*. Kota Kinabalu, Malaysia: IEEE, Nov. 2020, pp. 88–92.

[4] T. Deelmann, "Does Digitization Matter? Reflections on a Possible Transformation of the Consulting Business," in *Digital Transformation of the Consulting Industry*, V. Nissen, Ed. Springer, 2018, pp. 75–99.

[5] A. Emrich, S. Klein, M. Frey, and P. Fettke, "A Platform for Data-Driven Self-Consulting to enable Business Transformation and Technology Innovation," in *Multikonferenz Wirtschaftsinformatik*, Lüneburg, Germany, 2018, p. 7.

[6] T. Greff, F. Winter, and D. Werth, "Digitale Geschäftsmodelle in der Domäne wissensintensiver Dienstleistungen - Stand der Forschung und Transfer in die Unternehmensberatung," in *MKWI - Multikonferenz Wirtschaftsinformatik*, Drews P., Funk B., Niemeyer P., and Xie L., Eds., vol. 2018-March. Leuphana Universitat Lüneburg, 2018, pp. 1316–1327.

[7] A. R. Hevner, S. Ram, and E. College, "Design Science in Information Systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[8] D. Jung, F. Glaser, and W. Köpplin, "Robo-Advisory: Opportunities and Risks for the Future of Financial Advisory," in *Advances in Consulting Research*, V. Nissen, Ed. Cham: Springer International Publishing, 2019, pp. 405–427.

[9] S. Lauesen, *Software Requirements: Styles and Techniques*. Addison-Wesley, 2002.

[10] V. Nissen and H. Seifert, "Die digitale Transformation der Unternehmensberatung," in *Dienstleistungen 4.0. Band 2: Geschäftsmodelle - Wertschöpfung - Transformation*, 2017.

[11] V. Nissen, A. Füßl, D. Werth, K. Gugler, and C. Neu, "On the Current State of Digital Transformation in the German Market for Business Consulting," in *Advances in Consulting Research*. Berlin: Springer, 2019, pp. 317–339.

[12] V. Nissen, J. Kuhl, H. Kräft, H. Seifert, J. Reiter, and J. Eidmann, "Creating a Digital Consulting Solution for Project Management Assessments at Dr. Kuhl Unternehmensberatung: Development and Initial Demonstration of a Fully Automated Asset-Based Consulting Approach," in *Digitalization Cases*, N. Urbach and M. Röglinger, Eds. Springer, 2019, pp. 225–247.

[13] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, Dec. 2007.

[14] T. Rosqvist, M. Koskela, and H. Harju, "Software Quality Evaluation Based on Expert Judgement," *Software Quality Journal*, vol. 11, no. 1, pp. 39–55, May 2003.

[15] H. Seifert and V. Nissen, "Virtualization of Consulting Services: State of Research on Digital Transformation in Consulting and Future Research Demand," in *Digital Transformation of the Consulting Industry*, V. Nissen, Ed. Springer, 2018, pp. 61–73.

[16] D. Werth and T. Greff, "Digitale Beratung - ein Modell für den Mittelstand," *IM+io Fachzeitschrift für Innovation, Organisation und Management*, p. 3, 2016.

[17] D. Werth, T. Greff, and P. Zimmermann, "Self-Service Consulting: Conceiving customer-operated digital IT consulting services," in *ACIS*, San Diego, 2016.

[18] R. J. Wieringa, *Design Science Methodology for Information Systems and Software Engineering*. Berlin: Springer, 2014.