

An Approach to Transforming Requirements into Evaluable UI Design for Contextual Practice - A Design Science Research Perspective

Matthias Walter

Chair of Information Systems, esp. IS in Manufacturing and Commerce, Technische Universität Dresden
Dresden, Germany
matthias.walter3@tu-dresden.de

Abstract—We contribute a methodical approach in the context of IS design science research to develop UI prototypes for evaluations in practice-oriented research. Based on previous research on improving IS support for early product cost optimization, we present and discuss our methodical approach to derive UI prototypes based on an evaluated requirements model. The objective of the outlined approach comprising different steps is to derive a clickable UI prototype that is feasible for further artifact evaluation within institutional environments. Together with experts from the practice of software engineering we iterated through the working steps of the elaborated approach to determine its feasibility to derive a prototype and moreover, generate visual examples for each step to improve the approach's comprehensibility. In addition to the description of the approach itself we point to significant hurdles that have arisen with the application of it in order to generate learnings for other research projects.

I. MOTIVATION

IN times of globalization, demand rises for agility, innovation, and quality. Furthermore, shortened product life cycles and an amplified variety of product models have increased pressure on product manufacturers [1], [2]. In order to keep up with the global competition as such, optimizing product costs throughout a product's life cycle has become a major driver for economic success. To ensure the long-lasting economic success of products in the upcoming decades, organizations have been attempting to optimize product costs for the overall product life cycle. This is especially true for the discrete manufacturing industry, where products like cars, trucks, airplanes, and high-tech machinery are assembled out of thousands of globally sourced components [1].

Figure 1 describes the cost situation in the life cycle of such products. Surprisingly, 90% of all product costs are determined before production starts – and, thus, in the phase of product development. Linking this fact to the idea of product cost optimization to ensure economic success, it becomes obvious that product development phases offer the most potential to optimize product costs. Despite this immense potential, there is a lack of information system

(IS) support for product cost optimization during product development [3], [4], [5].

In our long-term research project, we aim to improve IS support for product cost optimization during product development and, therefore, aid the industry in making use of the cost optimization potential. Due to the practical relevance of the research problem and the demand for new IS approaches within the industry [4], [5], we initiated a research project together with the software corporation SAP SE. The resulting research collaboration, which includes various international companies, is framed by a design science research (DSR) approach [6]. First, we worked out a detailed problem identification together with companies among the discrete manufacturing industry [4], [5]. To overcome the identified problems, we elaborated major implementation challenges and requirements, which were further transferred into a requirements model [1]. After an industry evaluation of the requirements model [1], we now aim to design possible solutions to improve the support of IS for early product cost optimization and, thus, use the potential of product development phases (Figure 1).

In order to exploit the full potential of our industry research collaboration, it is recommended to evaluate DSR artifacts in their natural setting [7]. For this purpose, it is necessary to elaborate instantiations of potential problem solutions that can be evaluated by industry experts in practice. The question is: how can this be done?

Although DSR is generally gaining popularity in IS research [8], there are only a few research contributions that provide guidance for the design of at least partially instantiated artifacts in the context of DSR [9]. Further literature analyses have shown that the majority of the designed artifacts in DSR research are of type method or model [8] and can, therefore, only provide limited guidance for our practice-oriented approach to design potential solutions. In addition, the need to develop further research methods that immerse researchers in practice environments has been highlighted in the literature [10].

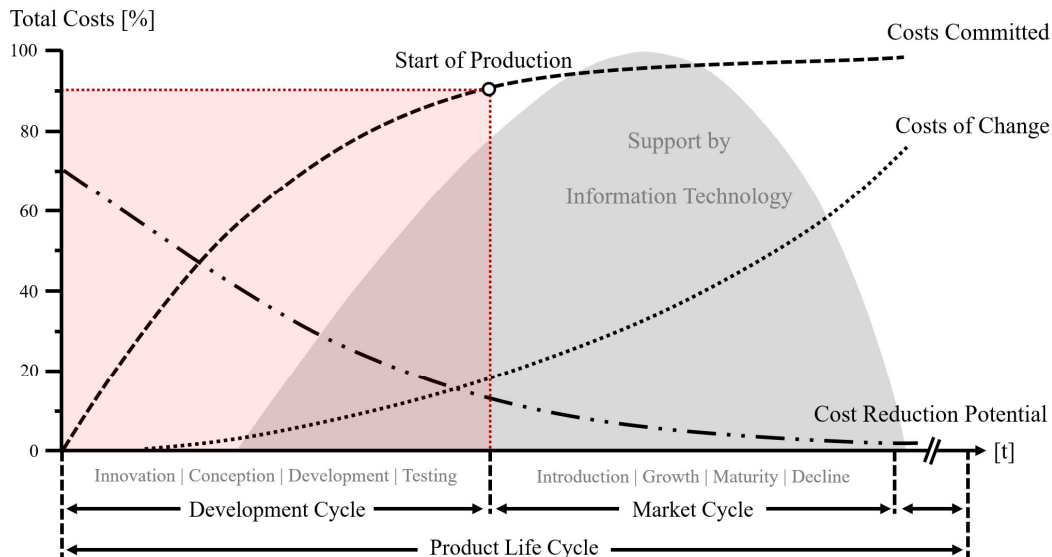


Fig. 1 Cost commitments and reduction potential along product life cycle [4], [11]

To improve this area of research, we would like to depict and discuss a methodical approach to develop user interface (UI) prototypes based on validated requirements. This approach has helped us in our industry collaboration research project, because such prototypes can be evaluated intuitively by experts without requiring a dedicated theoretical knowledge base. In addition, the development of UI prototypes requires considerably less efforts than regular software-based prototypes. Driven by the question of how to elaborate such evaluable UI designs, we focus the following research questions:

RQ1: Based on the elaborated requirements model, what could a methodical approach look like to transform requirements into UI prototypes that are suitable for iterative practical evaluations?

RQ2: What are the challenges of such an approach, and how can possible hurdles be mitigated?

To answer these research questions, we briefly introduce the research project and the research results to date. We then explain the elaborated approach to develop the UI prototype and explain the individual working steps in detail with the support of visual examples. The paper concludes with a discussion of results and an elaboration of lessons learned based on the application of this approach in the context of our research project, including possible pitfalls to provide valuable insights for other researchers.

II. BACKGROUND

Though there are IS available that aim at supporting product cost calculation during product development [12], a lack of functionality and the demand for enhancements has been identified [3]. Building on this, in our initial research steps, we identified missing support for early product cost

optimization and the resulting drawbacks for the industry [4], [5]. To establish a substantiated foundation for further research activities in a DSR context [13], we elaborated and evaluated a requirements model with experts from international companies within the discrete manufacturing industry [1]. Moreover, we identified major implementation challenges that have an impact on the solution acceptance in the industry and, therefore, will have an impact on the current phase of our research: solution design.

In theory, the DSR artifact design and development process is described as a rather individual and creative engineering process [9], [14]. Though there are general approaches available in the literature describing this process, the lack of guidance for artifact design in IS literature is evident [9].

Seeking such processual guidance for developing a potential solution, we had to consider an important circumstance in our research domain: Product cost optimization measures are derived in a rather unpredictable context based on deliberations of various stakeholders (e.g., product developers, production process engineers, production planners, cost controllers, and purchasers) and their collaboration [4]. Such deliberations do not follow best-practice patterns, but originate from an evolving knowledge-base along the product development phase, and therefore, result in dynamic, context-driven product cost optimization processes. Such processes can be classified as one type of emergent knowledge processes, as argued in our requirements elaboration [1].

Hence, potential solutions must support dynamically changing processes with a bandwidth of deliberations and tradeoffs based on complex, evolving expert knowledge bases within the organization, which is exclusive to product development [15].



Fig. 2 Degree of process specification [1], [16]

With this justificatory knowledge in mind, developing potential solutions becomes a challenge: First of all, it is important to determine the right degree of process support for experts using the IS without being too restrictive to limit their capabilities while carrying out product cost optimization across their organization. In addition, following the argumentation in [7] and [16], the developed solutions must be evaluated in the specific environment of their application. In our case, this requires an evaluation where business experts try to use our prototypes as support for their product cost optimization processes.

Due to the strong tie of our research problem to an organizational context, it is necessary to reflect whether the DSR methods to develop such a UI prototype fit our research context. This is especially important since it has been revealed that DSR methods consider organizational intervention to be of lesser importance [17]. In contrast to this, [10] argued to further develop specific methods aiming at the co-constitutional character of user contexts and institutional environments.

Such co-constitutional aspects can be approached in multiple ways. [18] criticized the strong sequencing of DSR, which separates developing artifacts from evaluating artifacts. To overcome this separation, action design research (ADR) has been recommended as a DSR method to closely link development to evaluation [18]. Further recommendations have been proposed in the literature [19]. Based on a comparative analysis of DSR with the constructive research approach, [19] highlighted the potential to improve DSR methods by developing best practices for collaborative development.

Beyond the DSR discourse, it is worth examining how IS prototypes are being used in practice. One area of regular prototype development and evaluation is agile software development (e.g., rapid prototyping) [20]. One specific example within the area of development research is mockup-driven development [21]. This approach makes use of UI prototypes (mockups) to receive early and continuous feedback to guide modeling and, thus, align further application development. This coincides with the intention for ADR as a method [18] to combine development and evaluation of artifacts into iterative steps.

We would like to take advantage of the experiences that have been made with mockup-driven development approaches to build a bridge between our scientific research and the evaluating group of practitioners. By this attempt, we intend to enable business experts within the discrete manufacturing industry to quickly grasp core intentions of

our prototypes without preceding knowledge transfer. Furthermore, less effort would be exerted to create UI mockups instead of software prototypes, allowing shorter artifact iterations. The use of UI prototypes as a feasible option for DSR artifact evaluation has already been shown in other DSR projects [22].

III. METHODOLOGICAL APPROACH

The purpose of this paper is to draft and discuss a processual approach to develop evaluable UI prototypes to enhance practical research collaborations. This practical research collaboration is important for the problem solution, as product costing has a strong focus on expert knowledge [5], [23]. Moreover, industrial practice is the most important source of information for cost optimization projects [24]. Therefore, the access to knowledge from practitioner communities is fundamental for our research [1]. At the same time, such knowledge ensures that our research is relevant to practice [25].

Within our long-term design science research project, we elaborated and evaluated a requirements model [1] that included 30 detailed requirements in combination with implementation challenges on the basis of industry expert interviews and focus groups. In addition, the knowledge exchange with domain experts helped us to gain a holistic understanding of applied product cost optimization processes among different companies in an applied environment [5]. This was essential to develop a rather abstract approach – with respect to emergent circumstances toward the degree of process specification (see Background) – toward early product cost optimization, which has been agreed upon by industry experts (Figure 3).

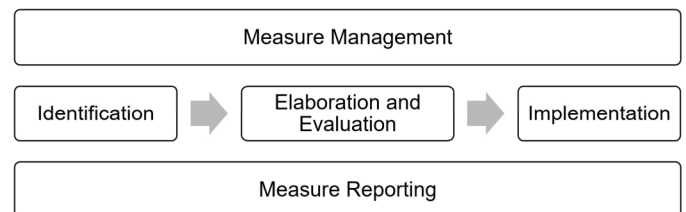


Fig. 3 Approach to support early product cost optimization [1]

As this approach provides support for the maturation of individual optimization measures (e.g., make-or-buy decisions, evaluation of alternative product concept or production processes, or optimization of logistic costs) [1], [5], our UI prototype development should be guided by such scenarios in which optimization measures mature from their identification through evaluation to implementation.

Therefore, we initiated our prototype development with the derivation of example optimization processes (Figure 4). These process descriptions were then used to deduce different user scenarios, each representing one step within the exemplary optimization process. Such a user scenario

describes the sequence of working tasks from the perspective of a specific expert user involved in the optimization processes (e.g., product cost controller, purchaser, engineer).

Such user scenarios are designed in a way that they consider aspects of the previously evaluated requirements. This is important because we used these scenarios for a focus group with five UI experts from our research partner SAP SE who were not familiar with our research domain of product cost optimization. Since the UI experts were not familiar with the industry processes, we were able to validate and adjust our user scenarios in terms of comprehensibility. After the establishment of a common understanding, it was the task of the UI experts to transfer the user scenarios independently into UI drafts without further assistance. This was done using a paper-based approach.

Afterwards, each UI expert presented his UI drafts for each user scenario to the whole group. During this presentation, each focus group participant was encouraged to provide feedback for the different UI drafts. This feedback was helpful for us to then transfer the various paper-based designs into a digital “best-of-breed” UI prototype combining the most valuable concepts. This prototype was created with Balsamiq mockup software [26], which allows individual UI screens (and their elements) to be linked into a “clickable UI prototype.”

As the last step of this development approach, we internally evaluated the clickable UI prototype first against our user scenarios and example processes, and later against the evaluated requirements in detail. This iteration allowed us to adjust the prototype design and the user scenarios.

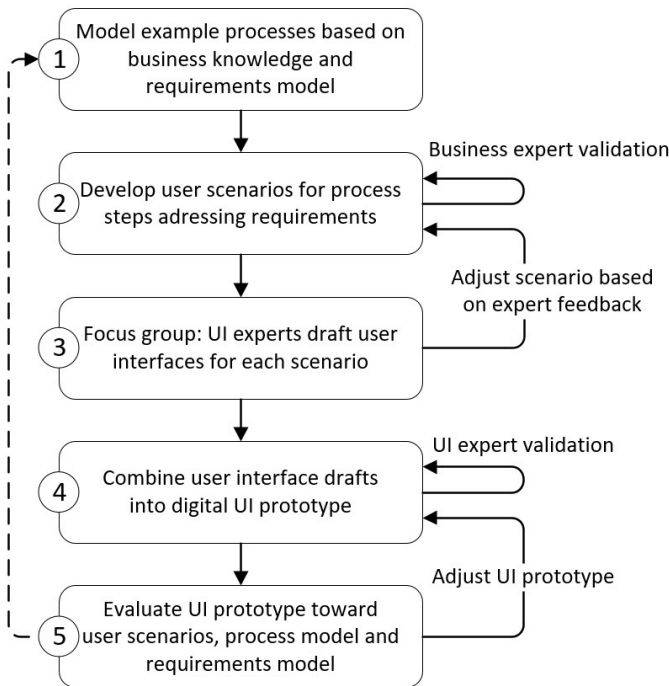


Fig. 4 Approach to transform requirements into UI prototype

This iterative approach to design a rather mature clickable UI prototype is necessary, because practitioners within the industry are usually limited toward their buy-in into research activities (especially timewise) [27]. For this reason, it makes sense to integrate these important knowledge resources in less granular and well-prepared research increments, being feasible for practice-oriented evaluation to maintain collaboration motivation.

In the next section, we will discuss the single steps of the development approach in detail and support them with examples from our research project.

IV. PROCESS DETAILS

A. Processes Modeling (1) and User Scenario Design (2)

According to [10], the process of designing a DSR artifact always has its point of departure in the current world. This is especially true for our practical problem of lacking IS support during early product cost optimization. Therefore, we first established an understanding of relevant optimization approaches [5] and transformed these into exemplary process flows. We started with a rather informal descriptive approach, and later used model-based approaches to establish a basis for a collective understanding among the stakeholders. In detail, we modeled the exemplary process flows with BPMN 2.0 (Business Process Model and Notation) [28].

Due to the emergent process characteristics of our research domain, process modeling is a challenge (see Background). This is because of emergent process adaptations in practice that can neither be foreseen nor fully defined by us; yet, at the same time, it is immanent that there is no chance to reach full process coverage. Therefore, we decided to transform those optimization approaches into process models, which have been evaluated as the most relevant for early product cost optimization within the discrete manufacturing industry (Table 1) [5].

TABLE I.
TOP 5 OPTIMIZATION APPROACHES (RATING SCALE 0 - NOT IMPORTANT TO 10 - VERY IMPORTANT) [5]

Optimization Approach	Avg. Rating
Make-or-buy analysis	7.83
Material price optimization	7.78
Alternative concept and product designs	7.33
Alternative production plants	7.29
Alternative reference components, assemblies, materials, and recipe ingredients	7.06

It is important to emphasize that this set of modeled exemplary processes does not claim to be complete. Nevertheless, it provides valuable insights into where and

how the support of practitioners needs to be improved with IS.

Furthermore, process modeling helped us to identify the various process stakeholders. By differentiating between different user roles and their individual tasks within the optimization process, we were able to identify each business expert's contribution to mature a cost optimization measure from a vague idea to its implementation into a cost calculation. Building on this, we derived specific user scenarios, each representing one step within the modeled optimization process. These user scenarios form a descriptive sequence of granular working tasks from the perspective of the specific user role. These user scenarios were then enriched by transforming evaluated requirements into dedicated user actions along the processual working tasks. This was done similarly to the approach of formulating user stories in agile software development to describe functional requirements [29].

User scenario: Create a new proposal for a make-or-buy analysis for a specific component in your product

The cross-functional “cost-optimization workshop” for the development of product “Pump P-100” is in progress. You have identified a target cost deviation for the component “Casing” in your current cost calculation. Together with experts from different specialties, like purchasing, engineering, and production planning, you have identified and discussed potential optimization measures. As product cost controller, Peter, it is your task to initiate the most promising optimization measure “Make-or-buy analysis for component Casing” for further evaluation.

To do so, the following working steps must be accomplished:

- Create the measure for component “Casing” in your costing structure.
- Create an achievement plan for the measure. The cost calculation of “Pump P-100” is part of a customer quotation and, therefore, must be done by 2017-11-30.
- In order to reach the target costs, the measure must achieve savings of 300€. This targeted impact was agreed on during the cost optimization workshop. Maintain this targeted impact for the measure to enable further evaluations.
- Assign the responsibility for next evaluation steps to user Joe from purchasing department.

Fig. 5 User scenario to create a new optimization measure

Figure 5 shows a rather simple scenario from the beginning of the optimization process for a measure of type “make-or-buy analysis” (Table 1) that is being used in our prototype development. The scenario is written from the perspective of the product cost controller – who is responsible for the management of cost optimization measures at most companies [4] – and addresses requirements from our evaluated requirements model in [1]. For example, we approach the requirement toward a functionality that supports “Target costing” during the identification of an optimization measure (“Identification,” Figure 3). In addition, we address requirements from the area of measure management (“Measure Management,” Figure 3). In detail, this is the basic requirement of a centralized platform to manage the optimization measure

among cost calculation projects, the assignment of responsibilities, the ability to create achievement plans, and the need to link the measure to components in existing cost calculations. To prioritize requirements toward our prototype, the requirements were not only validated toward their general relevance, but were prioritized on a scale from 0 (not important) to 10 (very important). Table 2 shows selected requirements and their evaluation results.

TABLE II.
EXCERPT OF EVALUATED REQUIREMENTS FROM REQUIREMENTS
MODEL [1]

	Avg. Rating	Std. Dev.
Measure Management		
Collect cost-optimization measures	7.78	1.72
Select measures for [...], and cost items	8.06	1.43
Define responsibilities	6.78	2.44
Create achievement plans	7.06	2.29
Estimate measure impact	8.29	1.45
Measure Identification		
Target costing	8.72	1.45

The user scenario in Figure 5 implies that there is an IS for performing early product cost calculation in place which our prototype can integrate with (see Background). As identified in previous research, this is not always the case in practice [4]. Nonetheless, the elaborated implementation challenges in previous research demand an integrated approach to ensure artifact acceptance by the end-users [1]. Moreover, this scenario implication seems to be valid because such IS are available on the market [12], although their functionality cannot cover all requirements among the discrete manufacturing industry [5].

Moreover, we introduced avatars for each user role (Figure 5) in our optimization processes. Through this idea adopted from a gamification concept for agile software development [30], we aim at making the descriptive, mostly text-based user scenarios, more attractive to people being involved in our prototype development and evaluation process (e.g., UI experts or evaluating domain experts). In addition, this should improve transparency about available user roles and their dedicated tasks in practical environments. These avatars remained consistent among the different optimization scenarios and the prototype development iterations (Table 1).

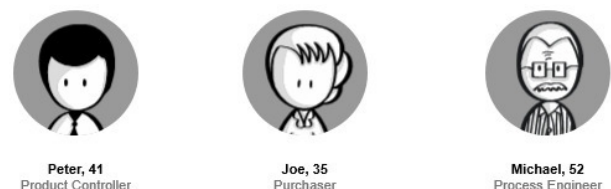


Fig. 6 Avatars that were used for the make-or-buy optimization process

B. Focus Groups with UI Experts (3)

The elaborated user scenarios build the foundation for the next step in our prototype development process. In contrast to software engineering practice where mockups are being used to enrich descriptive requirements [31], our research focuses on the derivation of design principles to improve IS support for emergent cost-optimization processes. Therefore, we decided to use the process of transforming requirements into visual mockups as an initial step to identify possible success criteria and design principles to support ex-post evaluations [32].

Due to our research collaboration with SAP SE, we had the chance to involve experts for user interfaces design in business software from SAP Innovation Center Network [33]. In a focus group involving five senior experts, who have focused on UI conception and development, with more than 7 years of work experience each, we firstly introduced the research topic driven by our problem identification [4]. Afterwards, we introduced the research approach (Figure 4) and stated the objective to create UI drafts aiming at the task fulfillment described in the set of user scenarios. As the last step of our introduction, we outlined the selected optimization process based on our BPMN models with the support of user role avatars.

Since the role of creativity to suggest solutions was highlighted in [9], we chose a pure paper-based approach to initially draft UI proposals. Nevertheless, this deliberation was not made without intention: Research has shown that paper-based approaches have certain advantages over digital prototyping approaches. Paper-based prototypes are particularly preferable when different design solutions need to be negotiated and stakeholder feedback is considered important [34]. Since we planned to have open feedback discussions among experts and, moreover, wanted to combine different solutions into a “best-of-breed” UI prototype later in the process, the paper-based approach seemed more valuable from a research perspective.

To create the UI drafts, we iterated through the optimization process based on the user scenarios as an iteration increment. The user scenario was presented to the focus group following the opportunity to clarify questions regarding its comprehensibility. By challenging the comprehensibility with the experts, we were able to further enhance the user scenarios. After reaching a collective understanding of the user scenario, the experts were asked to draft the UI individually. To underpin our claim to exemplary action, selected results of this focus group session are presented in Figure 7.

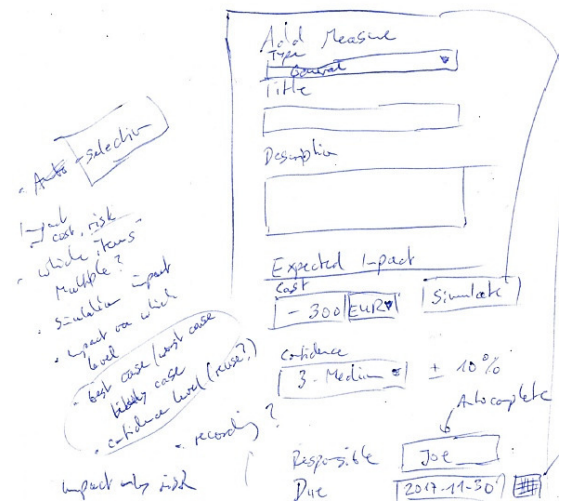


Fig. 7 Paper-based draft implementing user scenario from Figure 5

At the end of each iteration cycle, each focus group participant presented his UI draft for the specific user scenario, supported by an argumentation for the chosen UI concept. Afterwards, focus group members could provide feedback and clarify open questions. In total, we derived 47 individual, paper-based UI screen proposals for the optimization process of type “make-or-buy analysis” distributed over two separate sessions with a total duration of five working hours. The challenge in the following development step is to combine the various drafts to an evaluable prototype.

C. Development (4) and Validation (5) of a UI prototype

To transform the various paper-based UI drafts into a UI prototype that is suitable for evaluation with industry experts, we combined the most promising (also, in regard to UI expert feedback during the focus group session) drafts into digital mockups. As mentioned earlier (see Development Approach), we used the dedicated mockup software Balsamiq [26] to create a digital UI prototype.

Such a digital UI prototype consists of multiple mockups, each representing a certain UI screen (Figure 8). The major advantage of such a digital prototype over a paper-based prototype is the ability to link the different mockups (UI screens and their elements, such as buttons or text fields) to each other. The result is a UI prototype that a user can click through similarly to a real, implemented prototype.

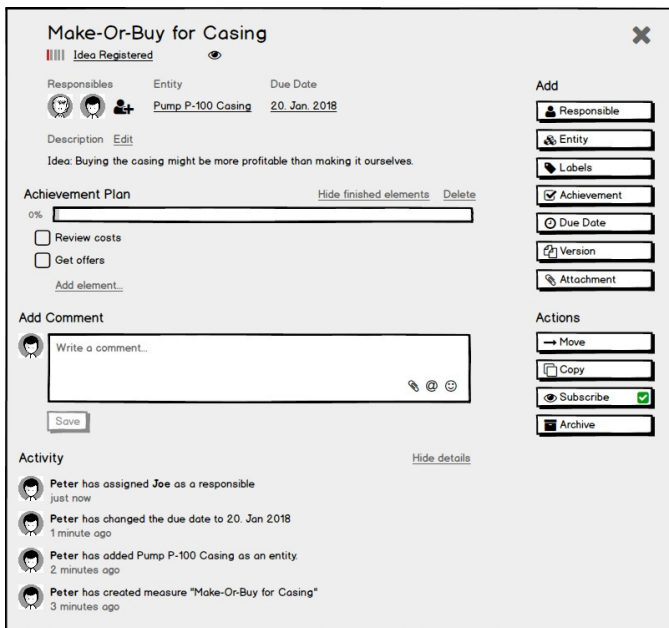


Fig. 8 UI mockup for user scenario from Figure 5

As was to be expected, not all expert UI drafts could be combined with each other. This was not due to specific UI elements or their arrangement in screens, but rather in terms of different conceptual approaches to address user scenarios. With the help of the digital UI prototype, we can easily exchange a series of UI screens to provide alternative concepts for parallel evaluation. This is extremely helpful to simulate different approaches toward process support (see *Background*).

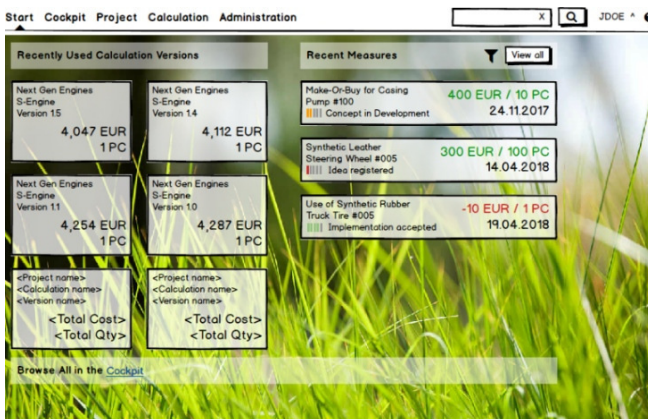


Fig. 9 Mockup of SAP Product Lifecycle's homescreen extended by functionality toward early product cost optimization

To stress the argumentation of [10] once more (see *Processes Modeling (1) and User Scenario Design (2)*) and, thus, enable an evaluation in institutional environments, we designed the clickable UI prototype to integrate with mockups of SAP Product Lifecycle Costing (Figure 9) [35]. This software is a dedicated solution to support early product cost calculation (see *Background*) and is one possible software solution available on the market that could

be integrated with our UI prototype [12]. At the same time, this visual integration was necessary to address one of our elaborated implementation challenges demanding an integrated approach for optimization measure management [1].

The further development of this clickable UI prototype is linked to its internal evaluation. This internal evaluation uses three previously elaborated elements for evaluation: process models, user scenarios, and the requirements model, including a requirement description (Figure 10). Initially, we verified that the prototype addressed the tasks outlined in the user scenarios. Furthermore, we checked whether the prototype implemented by the user scenarios still corresponded to the requirements description in the validated requirements model. Finally, the prototype's click sequence needed to support the order of the tasks according to the process models.

Define responsibilities

Due to highly interdisciplinary activities, measures' responsibilities, such as functional organizations or individual task owners, must be distinctly assigned to each measure. It ensures clear organizational responsibilities for the cross-functional optimization processes.

Create achievement plans

Due to time pressure during product development, product cost optimization must be accomplished in time. Therefore, measures must contain either due dates or product development-related gates, enabling ongoing reporting.

Fig. 10 Exemplary descriptions for the evaluated requirements (Table 2) [1]

After the successful internal evaluation, we completed the development of the UI prototype that is intended for further evaluation by domain experts. This evaluation is planned in two parts: First, we let domain experts, such as consultants, product owners, and solution owners for product costing solutions, from our research partner SAP SE do the evaluation. Based on their feedback, we iterate through the outlined development process (Figure 4) once more. Business experts from the industry will then take over further evaluation.

V. LESSONS LEARNED

In addition to the outline of our approach to transform requirements into a UI prototype, we want to contribute lessons learned from the application of this approach for our research project. Although the approach with its examples from our research project seems easy to implement, there are still some hurdles to overcome. We want to show these in the following, as they can also be relevant for other researchers in similar research contexts.

During the focus group sessions with UI experts (see Focus Groups with UI Experts (3)) we underestimated the duration of the iterations per user scenario. Looking back,

the time required for each scenario can be estimated to be one hour. Contrary to our expectations, the initial scenario explanation with subsequent questions toward comprehensibility was very time-consuming. This was mainly due to the UI expert's lack of knowledge in the specific research domain of early product cost optimization and its application in practice. Due to the unexpected delay, we had to schedule a second appointment with the UI experts to finish the draft of the optimization process of type "make-or-buy" (see Processes Modeling (1) and User Scenario Design (2)). Due to time constraints, we had to consult another UI expert and, thus, employed initial ramp up efforts once more. In order to counter such timing problems early, sufficient time should be planned for the appointments.

From our perspective, there is a chance to speed up the drafting process. Our assumption of providing the user scenarios for each step of the process in terms of process comprehensibility led to a certain redundancy across several user scenarios (e.g., different user roles were accessing the collection of optimization measures in a similar way). By removing such redundant scenarios and, moreover, focusing on less self-explaining scenarios (e.g., the cockpit screen, Figure 9), we could have been more efficient without losing relevant findings for our research problem (see Background).

For the next prototype development, we aim at a focus group approach that further enhances the creativity of involved UI experts. Though our idea of a paper-based approach was appropriate in our research context to develop potential approaches solving identified problems, we strongly believe that user scenarios provide too much guidance to elaborate ground-breaking findings. Nonetheless, the approach to draw paper-based prototypes is less time-consuming than drafting digital prototypes with dedicated software (See Development (4) and Validation (5) of a UI prototype).

In general, the approach to aim at such "clickable UI prototype" for evaluation is valuable for us. Instead of the need to implement a full-stack prototype, which, in addition, could have been limited by the extensibility options of available product costing software (see Processes Modeling (1) and User Scenario Design (2)), we could efficiently iterate through the development process to further enhance the prototype. This would simplify the testing and evaluation of several solutions in parallel. It would also motivate stakeholders to participate in the research (e.g., with new ideas or concepts), as development and its output would be progressing rapidly.

In addition, it should be mentioned that difficulties may arise when deriving and interpreting the process models and their user scenarios from the industry context into the research context. Therefore, we highly recommend

evaluating the single process step results (Figure 4), like process models or user scenarios with domain experts to prevent the derivation of (partly) incorrect scenarios. To easily communicate and evaluate such results, model-based approaches like BPMN 2.0 (see Processes Modeling (1) and User Scenario Design (2)) should be used to establish a collective understanding for all stakeholders.

Furthermore, it must be ensured that the transfer of research domain content to the UI experts has been successful. Though we thoroughly introduced our research domain, including the core of our problem identification [4], [5] and the approach to derive a UI prototype (Figure 4), multiple questions for each scenario were raised. This was time consuming, but necessary for a collective understanding. The transfer from paper-based UI drafts and their underlying conceptual ideas to the digital UI prototype involves similar transition difficulties, which we recommend verifying with UI expert consultations, at least briefly, after the mockups have been created.

VI. DISCUSSION

First, the question must be asked regarding whether the outlined approach is a valid contribution to practice-oriented DSR since we utilize a variety of well-established elements and approaches from software engineering (e.g., paper-based prototypes or mockup-driven development). According to [14], the development of tentative designs to solve identified problems is a rather pedestrian process in which no novelty beyond the state-of-art is required. Rather, the novelty should be part of the solution design itself. Following this argumentation, we do not state the UI prototype development approach as our research artifact, but as a valuable contribution to the research community.

This is especially true because, for example, the rare process guidance for artifact design has been criticized [9]. In detail, recent literature reviews have indicated only a small amount of DSR dealing with the development of software-based artifacts [8]. This is underpinned by [10], who requested methods to enable IS researchers to become immersed into institutional environments. As our approach results in UI prototypes for evaluation in practical contexts, we are convinced that it is a valuable and engaging approach toward practice-oriented DSR for all stakeholders.

What is indeed beneficial for our contextual prototyping are insights from works on action design research (ADR) as proposed in the context of DSR [18]. The interference with ADR and, therefore, the strong practical context opens room for discussions about the methodological rigor of our proposal to design a potential solution for further evaluation in the context of our problem to improve IS support for early product cost optimization. As argued by [36], there are certain conflicts in the discipline of IS research when it comes to the influence of methodical rigor. With this

contribution, we want to respond to and later solve a practical problem, which IS routine problem solving [37] has failed to address for many decades [3], [4], [5], [23], and now must be tackled by IS research.

Though we have not yet demonstrated the capability of the results to solve our identified problem [4], [5] as recommended by [6], the sole idea of designing an evaluable UI prototype has advanced our research project. The iterative validations and adjustments during the execution helped us to sharpen the understanding of the research; furthermore, the approach enabled new and improved existing collaborations with research stakeholders. According to [38], participation has the potential to enrich descriptions of the research process as well as increase the understanding of the artifact and its instantiation. Though the participation of experts in our approach led to unexpected efforts (see Lessons Learned), we can confirm and highly recommend a broad participation – especially if it concerns participants from other disciplines. Hence, the presented approach provides thought-provoking impulses and prevents tunnel vision among researchers.

This links to the contextual nature of our UI prototype for future evaluation. Earlier in this paper, we highlighted the emergent character of product cost optimization processes and the need to provide a context-integrative solution (see Background). The integration into the context of an industry application was possible and, thus, addresses one major implementation challenge to improve IS support in early product cost optimization [1]. Moreover, the evaluation of such UI prototype in practical contexts has already been shown in other research projects [22]. Therefore, we are convinced that the presented approach can contribute to finding answers to the research questions associated with our long-term research project. In addition, the concept to design the prototype iteratively improves result quality and, hence, helps to better use the already difficult-to-reach experts within the industry.

However, researchers who want to adapt this approach for their research must bear in mind that some hurdles may arise, especially regarding the transitions between the individual steps of the approach (Figure 4):

- Derivation of misleading processes and user scenarios
- Knowledge transfer to UI experts regarding the research domain and the objective to think beyond boundaries
- Transformation of UI drafts back into the research domain-oriented context

To avoid such issues, validation and participation are key contributions to successfully derive a ready-to-evaluate UI prototype.

VII. CONCLUSION

Overall, we outlined our approach to develop a clickable UI prototype based on an evaluated requirements model with 30 individual requirements in the context of our long-term research project. This research project follows a DSR process recommendation [6] to improve early product cost optimization in the discrete manufacturing industry. In conjunction with this practical problem and its emergent character [1], we need to design and propose a IS solution that can be evaluated in the context of institutional environments as part of an iterative evaluation. This evaluation is part of the next research step in our long-term project, and has started with the UI prototype derived out of the outlined approach.

The approach of transforming requirements into a UI prototype has proven its value for us: Well-established techniques and state-of-the-art approaches from the software engineering discipline helped to improve research quality and strengthen relations with stakeholders with reasonable development efforts (compared to software-based prototypes). The collaboration with and the contribution of UI experts to our research were especially appreciated.

Moreover, we address the evident need within the literature to provide further guidance for the design phase of DSR. To provide meaningful guidance to IS researchers, the presented approach is enriched with examples and increments from our research project. In addition, the *Lessons Learned* section should help to adopt the process more easily. By this, we hope to further encourage researchers in practice-oriented research, and, at the same time, motivate experts among industry to join scientific research projects to solve relevant problems.

Applying the recommendation from [6] to our research project, we further concentrate on the improvement of our UI prototype and its ability to solve the problem identified in [4] and [5]. This prototype improvement is mainly driven by iterative evaluations of experts from the industry.

ACKNOWLEDGMENT

User role avatars were created with Scenes™ by SAP AppHaus (<http://experience.sap.com/designservices/scenes>).

REFERENCES

- [1] M. Walter, C. Leyh, and S. Strahinger, "Toward early product cost optimization: requirements for an integrated measure management approach," in *Proc. of the Multikonferenz Wirtschaftsinformatik 2018 (MKWI 2018)*, Lueneburg, 2018, pp. 2057–2068.
- [2] I. Roda and M. Garetti, "TCO evaluation in physical asset management: benefits and limitations for industrial adoption," in *Proc. on APMS 2014: Advances in Production Management Systems*, B. Grabot, B. Vallespir, S. Gomes, A. Bouras, and D. Kiritsis, Eds. Berlin: Springer Berlin Heidelberg, 2014, pp. 216–223, https://doi.org/10.1007/978-3-662-44733-8_27.
- [3] G. Schicker, F. Mader, and F. Bodendorf, "Product lifecycle cost management (PLCM): Status quo, Trends und Entwicklungsperspektiven im PLCM – eine empirische Studie," *Arbeitspapier Wirtschaftsinformatik II (2/2008)*, Nürnberg: Universität Erlangen-Nürnberg, 2008.

- [4] M. Walter and C. Leyh, "Knocking on industry's door: product cost optimization in the early stages requires better software support," in *2017 IEEE 19th Conference on Business Informatics (CBI)*, IEEE: Thessaloniki, 2017, pp. 330–338, <https://doi.org/10.1109/cbi.2017.33>.
- [5] M. Walter, C. Leyh, and S. Strahinger, "Knocking on industry's door: needs in product-cost optimization in the early product life cycle stages," *Complex Systems Informatics and Modeling Quarterly (CSIMQ)*, Issue 13, pp. 43–60, 2017, <https://doi.org/10.7250/csimq.2017-13.03>.
- [6] K. Peffers, T. Tuunanen, C. E. Gengler, M. Rossi, W. Hui, V. Virtanen, and J. Bragge, "The design science research process: a model for producing and presenting information systems research," in *Proc. of the 1st International Conference on Design Science in Information Systems and Technology*, Claremont, 2006, pp. 83–106.
- [7] R. Baskerville, "What design science is not," *European Journal of Information Systems*, vol. 17, no. 5, pp. 441–443, 2008, <https://doi.org/10.1057/ejis.2008.45>.
- [8] R. Thakurta, B. Müller, F. Ahlemann, and D. Hoffmann, "The state of design – a comprehensive literature review to chart the design science research discourse," in *Proc. of the 50th Hawaii International Conference on System Sciences (HICSS)*, Waikoloa Village, Hawaii, 2017, pp. 4685–4694, <https://doi.org/10.24251/hicss.2017.571>.
- [9] P. Offermann, O. Levina, M. Schönherr, and U. Bub, "Outline of a design science research process," in *Proc. of the 4th International Conference on Design Science Research in Information Systems and Technology (DESIRIST '09)*, New York, NY: ACM, 2009, pp. 1–11, <https://doi.org/10.1145/1555619.1555629>.
- [10] K. Riemer and S. Seidel, "Design and design research as contextual practice," *Information Systems and e-Business Management*, vol. 11, no. 3, pp. 331–334, 2013, <https://doi.org/10.1007/s10257-013-0223-2>.
- [11] M. Eigner and R. Stelzer, *Product Lifecycle Management: Ein Leitfaden für Product Development und Life Cycle Management*, 2nd ed. Heidelberg: Springer, 2009, <https://doi.org/10.1007/b93672>.
- [12] S. Voelker, M. Walter, and T. Munkelt, "Improving product life-cycle cost management by the application of recommender systems," in *Proc. of the Multikonferenz Wirtschaftsinformatik 2018 (MKWI 2018)*, Lüneburg, 2018, pp. 2019–2030.
- [13] R. Braun, M. Benedict, H. Wendler, and W. Esswein, "Proposal for requirements driven design science research," in *Proc. of the 10th International Conference on Design Science Research in Information Systems (DESIRIST)*, B. Donnellan, M. Helfert, J. Kenneally, D. VanderMeer, M. Rothenberger, and R. Winter, Eds. Cham: Springer International Publishing, 2015, pp. 135–151, https://doi.org/10.1007/978-3-319-18714-3_9.
- [14] V. K. Vaishnavi and W. Kuechler, *Design Science Research Methods and Patterns: Innovating Information and Communication Technology*, 2nd ed. Boca Raton, FL: CRC Press, 2015, <https://doi.org/10.1201/b18448>.
- [15] M. L. Markus, A. Majchrzak, and L. Gasser, "A design theory for systems that support emergent knowledge processes," *MIS Quarterly*, vol. 26, no. 3, pp. 179–212, 2002.
- [16] M. Böhringer, "Emergent case management for ad-hoc processes: a solution based on microblogging and activity streams," in *Proc. of BPM 2010: International Conference on Business Process Management*, M. Zur Muehlen and J. Su, Eds. Berlin: Springer Berlin Heidelberg, 2010, pp. 384–395, https://doi.org/10.1007/978-3-642-20511-8_36.
- [17] R. Cole, S. Purao, M. Rossi, and M. K. Sein, "Being proactive: where action research meets design research," in *Proc. of 24th International Conference on Information Systems*, D. Avison, D. Galletta, and J. I. DeGross, Eds. Las Vegas, 2005, pp. 325–336.
- [18] M. K. Sein, O. Henfridsson, S. Purao, M. Rossi, and R. Lindgren, "Action design research," *MIS Quarterly*, vol. 35, no. 1, pp. 37–56, 2011, <https://doi.org/10.2307/23043488>.
- [19] K. A. Piirainen and R. A. Gonzalez, "Seeking constructive synergy: design science and the constructive research approach," in *Proc. of the 8th international conference on Design Science at the Intersection of Physical and Virtual Design*, J. vom Brocke, R. Hekkala, S. Ram, and M. Rossi, Eds. Berlin: Springer Berlin Heidelberg, 2013, pp. 59–72, https://doi.org/10.1007/978-3-642-38827-9_5.
- [20] A. Pranam, *Product Management Essentials: Tools and Techniques for Becoming an Effective Technical Product Manager*. Berkeley, CA: Apress, 2018, <https://doi.org/10.1007/978-1-4842-3303-0>.
- [21] J. M. Rivero, J. Grigera, G. Rossi, E. R. Luna, F. Montero, and M. Gaedke, "Mockup-driven development: providing agile support for model-driven web engineering," *Information and Software Technology*, vol. 56, no. 6, pp. 670–687, 2014, <https://doi.org/10.1016/j.infsof.2014.01.011>.
- [22] D. Lück and C. Leyh, "Enabling business domain-specific e-collaboration: developing artifacts to integrate e-collaboration into product costing," in *Proc. of the 12th International Conference on Design Science Research in Information Systems*, A. Maedche, J. vom Brocke, and A. Hevner, Eds. Cham: Springer, 2017, pp. 296–312. Springer, Cham, https://doi.org/10.1007/978-3-319-59144-5_18.
- [23] D. Lück and C. Leyh, "Integrated virtual cooperation in product costing in the discrete manufacturing industry: a problem identification," in *Proc. of the Multikonferenz Wirtschaftsinformatik 2016 (MKWI 2016)*, Ilmenau, 2016, pp. 279–290.
- [24] M. Mörtl and C. Schmied, "Design for cost - a review of methods, tools and research directions," *Journal of the Indian Institute of Science*, vol. 95, no. 4, pp. 379–404, 2015.
- [25] M. Rosemann and I. Vessey, "Toward improving the relevance of information systems research to practice: the role of applicability checks," *MIS Quarterly*, vol. 32, no. 1, pp. 1–22, 2008, <https://doi.org/10.2307/25148826>.
- [26] Balsamiq Studios, LLC. <https://balsamiq.com/>, retrieved 23rd March 2018.
- [27] H. Österle and B. Otto, "Consortium research," *Business & Information Systems Engineering*, vol. 2, pp. 283–293, 2010, <https://doi.org/10.1007/s12599-010-0119-3>.
- [28] M. Chinosi and A. Trombetta, "BPMN: An introduction to the standard," *Computer Standards & Interfaces*, vol. 34, no. 1, pp. 124–134, 2012, <https://doi.org/10.1016/j.csi.2011.06.002>.
- [29] M. Cohn, *User stories applied: For agile software development*. Boston, MA: Addison-Wesley Professional, 2004.
- [30] P. Lombriser, F. Dalpiaz, G. Lucassen, and S. Brinkkemper, "Gamified requirements engineering: model and experimentation," in *Proc. of the 22nd International Working Conference on Requirements Engineering*, M. Daneva and O. Pastor, Eds. Cham: Springer, 2016, pp. 171–187, https://doi.org/10.1007/978-3-319-30282-9_12.
- [31] G. Reggio, M. Leotta, and R. Ricca, "A method for requirements capture and specification based on disciplined use cases and screen mockups," in *Proc. of the 16th International Conference on Product-Focused Software Process Improvement*, P. Abrahamsson, L. Corral, M. Oivo, and B. Russo, Eds. Cham: Springer International Publishing, 2015, pp. 105–113, https://doi.org/10.1007/978-3-319-26844-6_8.
- [32] J. Venable, J. Pries-Heje, and R. Baskerville, "FEDS: a framework for evaluation in design science research," *European Journal of Information Systems*, vol. 25, no. 1, pp. 77–89, 2016, <https://doi.org/10.1057/ejis.2014.36>.
- [33] SAP Innovation Center Network, SAP SE. <https://icn.sap.com/home.html>, retrieved 26th March 2018.
- [34] R. Sefelin, M. Tscheligi, and V. Giller, "Paper prototyping - what is it good for?: A comparison of paper- and computer-based low-fidelity prototyping," in *Proc. of CHI'03: Conference on Human Factors in Computing Systems*. Ft. Lauderdale, FL: ACM, 2003, pp. 778–779, <https://doi.org/10.1145/765985.765986>.
- [35] SAP Product Lifecycle Costing, SAP SE. <https://www.sap.com/products/product-lifecycle-costing.html>, retrieved 23rd March 2018.
- [36] R. B. Gallepe, "The tyranny of methodologies in information systems research," *SIGMIS Database*, vol. 38, no. 3, pp. 20–28, 2007, <https://doi.org/10.1145/1278253.1278258>.
- [37] S. Gregor and A. R. Hevner, "Positioning and presenting design science research for maximum impact," *MIS Quarterly*, vol. 37, no. 2, pp. 337–355, 2013, <https://doi.org/10.25300/misq/2013/37.2.01>.
- [38] S. Jönsson and K. Lukka, "There and back again: Doing interventionist research in management accounting," in *Handbook of Management Accounting Research*, C. S. Chapman, A. G. Hopwood, and H. G. Shields, Eds. Elsevier, 2006, pp. 373–397, [https://doi.org/10.1016/S1751-3243\(06\)01015-7](https://doi.org/10.1016/S1751-3243(06)01015-7).