2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering

Evaluation of the Simulated Application of the UCD-LSI Method

- The iPeople Case Study-

Ulrike Abelein, Barbara Paech
Institute of Computer Science, University of Heidelberg,
Im Neuenheimer Feld 326, 69120 Heidelberg, Germany
{abelein, paech}@informatik.uni-heidelberg.de

Michael Kern, Maria Woydich
Sovanta AG, X-House, Mittermaierstraße 31
69115 Heidelberg, Germany
{michael.kern, maria.woydich}@sovanta.com

***Abstract*— In previous work we showed in a systematic mapping study that there is no method to enhance user-developer communication (UDC) in the design and implementation phase of large-scale IT projects (LSI). We then defined the UDC-LSI method. It is substantial especially for newly designed methods to evaluate them within a real-world context. As it is difficult to find a company willing to apply an untested method, as a first step to full evaluation we present in this work a case study where we study the utility and acceptance of a simulated application of the UDC-LSI method. To make the simulation as real as possible we first thoroughly analyzed the as-is status of the iPeople project. Then we simulated an instantiation of the UDC-LSI method for the iPeople project and we evaluated this instantiation with project participants. The case study showed that it is possible to instantiate the method for the project under study. The evaluation confirmed a positive effect of the UDC-LSI method on system success (effectiveness), the feasibility and high acceptance of the method and a positive effort-benefit ratio (efficiency).**

***Index Terms*— User-Developer Communication, User Participation, User Involvement, Case Study**

I. Introduction

The overall share of empirical studies in computer science is still small. However, according to [1] a method "must be evaluated with respect to the utility provided for the class of problems addressed." Therefore this paper reports on a first case study the utility and usability of the UDC-LSI method in a real-world practical context. In previous work, we showed that there is no method that targets large-scale IT projects and has the goal to improve user-developer communication in the design and implementation phase with the intention to improve system success [2]. Thus, we designed the User-Developer-Communication in Large-Scale IT projects (UDC-LSI) method. Then we wanted to research the effects of a concrete instantiation of the method in a real-world large-scale IT project, but it is difficult to find a company willing to apply an untested method. As a first step to full evaluation we present in this work a case study where we study the utility (i.e. feasibility, effectiveness and efficiency) and usability (i.e. acceptance) of a simulated application of the UDC-LSI method in the iPeople project.

This paper is structured as follows. We first present the case study design with the research questions, case selection and the research method. Afterwards, we describe the simulated instantiation and show and discuss the results of the evaluation. We then discuss the threats to validity and conclude with a summary of the case study.

II. Case Study Design and Execution

We designed and conducted a case study according to instructions from Runeson [3]. The UDC-LSI method is defined for large-scale IT projects and its purpose is to increase system success through the increase of UDC in the design and implementation phase. Thus, we needed to identify a large-scale IT project with issues in UDC.

The *case company* sovanta AG is a fast growing company with currently about 60 employees. The main purpose of the *iPeople Business Application* is to support managers in personnel management. Thus, it presents Human Resource (HR) Key Performance Indicators (KPIs) to sales managers who meet monthly with their assigned branch managers. The application has been rolled out in 28 countries and it has 4500 users,

The iPeople project has an effort of about 750 person days, the project run time was 2 years plus 10 months and there are about 20 releases. The *project stakeholders* are the business and IT side. The business side is mainly represented in the project by one project manager (PM) from the customer. Each country has a key user (i.e. user representative) in the project. Unfortunately, we only had access to the IT personnel, but not to the business side within the case study. The IT personnel are the project sponsor (i.e. responsible for the budget), the PM from the company side, six developers, and one user interface (UI/UX) designer.

The project fulfills the criteria of our definition of a large-scale IT project: large amount of users, rollout in multiple countries, and project duration more than a year. The project is a customer-specific software development project, using a flexible, agile-like development, thus it does not use traditional methods. However, there are issues with UDC, i.e. the communication of the IT personnel with the customer PM and the key users. We believe that the described context is suitable as our case study context.

As preparation for the case study we thoroughly analyzed the as-is status of the project with regard to the current development process, established communication structures, revolving issues and user-relevant decisions. Therefore we studied the existing project artefacts and interviewed key

978-1-4673-7031-8/15 $31.00 © 2015 IEEE
DOI 10.1109/CHASE.2015.10

personnel. The analysis showed that there are issues in the development process and in the current communication structures with the users, i.e. currently all communication is document-based or mediated through the customer PM. Thus an improvement of the UDC process, including direct communication between users and IT personnel, is useful for the iPeople project. We identified 18 user-relevant decisions; which is an indication that there are topics and decisions that should be discussed with users. The central project management tool, used in this project, during development is the JIRA tool holding all the relevant documents. In particular each feature is represented in a ticket in JIRA. Since the iPeople project is the "oldest" project in the company, later developed, successful processes from the company are not used within this project. These successful processes include documentation of specification in a "scribble doc" (a document that contains for each feature a mockup of the designed screen, as well as descriptions of the provided functionality) and direct communication with users through workshop sessions. We partially included those processes in the project specific adaptation of the UDC-LSI method.

The conducted case study can be categorized as a *single case study with one unit of analysis*: The *iPeople project.* The *object of study* in this case study is the *UDC-LSI method*. As stated in [1] the utility (i.e. feasibility, effectiveness and efficiency) and usability (i.e. acceptance of the users) of a design artifact must be rigorously demonstrated via well-executed evaluation methods. We therefore raised the following research questions:

- RQ 1 (feasibility): Is it feasible to implement the UDC-LSI method in the unit of analysis?

The hypothesis H1 is, that the project participants consider is feasible to implement the UDC-LSI method in the project.

- RQ 2 (effectiveness): Does an implementation of the UDC-LSI method increase system success?

The hypothesis H2 is, that the project participants consider that the application of the UDC-LSI method has positive effect on system success.

- RQ 3 (efficiency): Is the effort of executing the method worthwhile its value?

The hypothesis H3 is that, the project participants consider the benefits of applying the UDC-LSI method outbalance the effort.

- RQ 4 (acceptance): How usable is the UDC-LSI method?

The hypothesis H4 is, that the project participants think the UDC-LSI method is usable.

As suggested in [3] we use a mixed method approach with various data sources to limit the effect of only interpreting data from one data source, i.e. existing project documentation, attendance of meetings, workshop sessions to define the new processes and interviews for the assessment. We also use methodological triangulation by combining qualitative methods (e.g. answers to open questions in interviews) and quantitative methods (e.g. questionnaires on Likert scales). The qualitative data from open questions is summarized in categories [3]. Furthermore we take into account viewpoints of different roles.

For treatment design we instantiated - based on the results of the as-is study - the UDC-LSI method for the iPeople project. In particular, we developed detailed examples of the method application in the project. The as-is study identified different user-relevant decision for discussion with the users. First there are design decisions based on the requirements of the users that have to be discussed in the design phase. Secondly, there are decisions to be discussed with the users in the implementation phase. Both differ in the documentation and communication needs. Therefore, we present in the following sections the instantiated process in two parts: one for the design and one for the implementation phase.

For the evaluation part we conducted nine fully structured interviews with project participants (1 project sponsor, 1 IT PM, 1 designer, 3 front end developers, 3 back end developers). The interviews were done in person. Eight interviews were done in German, one in English. We presented the process with the instantiated method and the example for each phase (design and implementation), and then asked for the project participants' opinion. We used closed questionnaires with Likert scale to ensure objectivity and comparability of the results. To ensure the right interpretation, we also included questions for rationales and open questions. For data analysis we recorded all interviews and transcribed the open questions. The answers of the open questions are translated into English and are summarized by counting similar answers. In total we recorded 737 min of interview time.

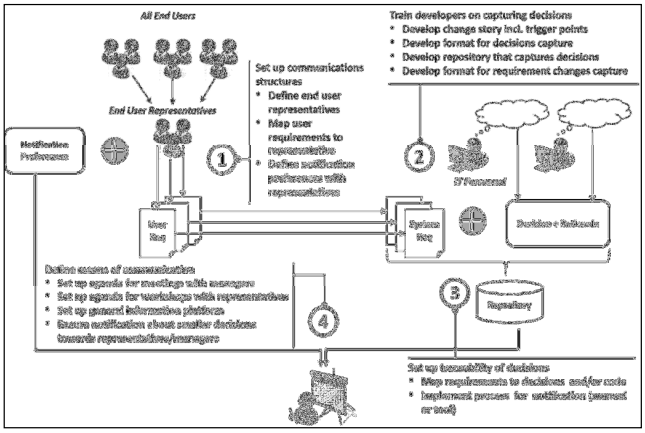

Fig. 1. UDC-LSI Method

III. Applying the UDC-LSI Method to Practice

The UDC-LSI method comprises four parts (setup of communication structure with user representatives, training of developers on the capture of decisions and changes, set up of the traceability of decisions, and definition of the means of communication), each with subtasks (see Figure 1). In order to instantiate the method, we start from the four parts, and map them to five criteria (see Table I). Methods define processes, which can be textual descriptions [1]. We therefore continued the instantiation of the UDC-LSI method through two detailed processes and a corresponding practical example of the iPeople project (see Figure 2). In the following we describe the instantiation rationale for the parts and the two processes with examples.

Part 1 – Set up of Communication Structures with User Representatives is important to understand and select user

representatives. It is comprised of three subtasks. The first subtask is to "define representatives for each 'class' of users", with a stakeholder analysis. In the project there are four roles in 28 countries, thus user representatives would be required for each role and country. However, we conducted a mapping of the usage profiles of all features by roles based on the iPeople product description. This mapping revealed that most features are used similarly by different roles. Therefore we decided that the already existing single key users per country are suitable user representatives (all 28 for the design phase, and 4-5 selected for the implementation phase). The second subtask is to "map user requirements to one or more user representatives", to ensure discussion of requirements/feature with the right user representative. Since we do not differentiate between different roles, this mapping is not necessary. The last subtask of this part is to "define notification preferences with the user representatives". Since we did not have direct access to the user representatives, we were not able to ask them about their notification preferences. Nevertheless, we define explicit triggers, where input of the users is needed for the IT personnel. We believe that the user representatives with their role to spread the iPeople solution within their countries are interested in all user-relevant decisions of the project.

TABLE I. INSTANTIATION OF THE UDC-LSI METHOD

	Criteria	Design phase	Implementation phase
Documentation	User-relevant decision	Design decisions how to implement requirements	Decisions based on new requirement, unclear specification or technical issue
	Format	Mockups and data definitions	New field within JIRA tickets (structured with question, alternatives, implication)
	Tool	Scribble doc	JIRA (existing project management tool)
	Traceability	Requirement –> feature -> wire-frame / mockup + data definition → scribble doc ID	Scribble doc ID
Communication	user repr...	28 existing key users per country	1. Level: customer PM 2. Level: 4-5 selected user representative
	Trigger	Completion of first concept in scribble doc 1.0	1. Level: In each weekly jour fixe 2. Level: at least 5 open decisions for clarification with users
	Means of communication	Workshop series with 28 user representatives; participation of 1 designer. 1 front end and 1 back end developer	1. Level: existing telephone conference; participation of 1 designer. 1 front end and 1 back end developer 2. Level: Workshop with 4-5 selected user representative; participation of 1 designer. 1 front end and 1 back end developer

Part 2 – Training of Developers on the Capture of Decisions and Changes is important to explain the new processes to IT personnel and to ensure the required infrastructure. It comprises four subtasks. The first subtask is to "develop a transformation story, incl. trigger points to start communication with users. We defined the following trigger points (see processes): For the design phase completion of first UI conception in a scribble doc, and for the implementation phase each weekly jour fixe (1st level) and additionally a trigger (2nd level), if there are 5 open decisions for clarification with users. Since we build upon existing processes it is not required to have a detailed training for developers. We believe one meeting with the project participants to explain them the new processes will be sufficient. The second subtask is to "develop a format for capturing of decisions". We reuse the existing process of the case company from other projects and thus suggest to communicate design decisions via mockups and extend them through data definitions.

The third subtask is to "define a format for changes in requirements". We observed that changes in requirements occur mainly within the design phase. In the implementation phase, there are more detailed decision required. We suggest a pragmatic solution for the scribble doc, which is just to highlight changes through formatting the new parts (bold, italic). For the implementation phase, we suggest to highlight the changes within JIRA tickets. The final subtask of this part is to "build up a repository for decisions". Since the process differs in design and implementation, we chose two different repositories. For the design phase the scribble doc is the tool, whereas within the implementation phase, we use the standard project management tool of choice, as IT personnel uses JIRA during development anyway. For information purposes, it is possible to do a report of JIRA and send that around as meeting minutes.

Part 3 – Set up of the Traceability of Decisions has only two subtasks. The first subtask is to "map each decision and change to a requirement", which differs for the phases design and implementation. In design mapping is done by guiding users through mockups, i.e. conceptualization of a requirement. Thus, a mapping from requirement to feature is done implicitly. Each feature gets an ID in the scribble doc and a mockup and the data definition describes the feature. In the implementation phase, we reuse the scribble doc ID to enable a communication based on visual representations towards the users. The second subtask is to "implement the notification process for users". Currently, there is no communication about design decisions with users. For the design phase, we use the already established process in other projects with two workshops: one to discuss and one to align the mockups with users. During implementation phase, we replace the ad-hoc communication with customer PM by a structured two level process, i.e. the 1st level is a discussion with the customer PM and the 2nd level is a workshop with user representatives. For both phases we defined triggers.

Part 4 – Definition of the Means of Communication has three subtasks. The first subtask is to "set up a fixed agenda for meetings with managers". Currently, there is one monthly manager meeting, however when we analyzed typical user-relevant decisions of the iPeople project, most did not need that escalation level. Therefore, we do not suggest manager meetings for UDC in the iPeople project. However, in case of an escalation this meeting can still be used. The second subtask is to "have workshops with the user representatives". There is no communication with the user, but the IT personnel is interested in getting user feedback. Since the workshop concept is already used, we suggest reusing that concept (i.e. Design - workshop series with 28 user representatives; participation of 1 designer. 1 front end and 1 back end developer, and Implementation – 2 level process with customer PM/selected user representatives). Currently these workshops are moderated only by designers. However, we believe that is important to include all roles, i.e. include developers and IT PM from IT personnel. The last subtask of the method is "setting up a general information platform". This part has not been explicitly instantiated within the iPeople project, however both suggested

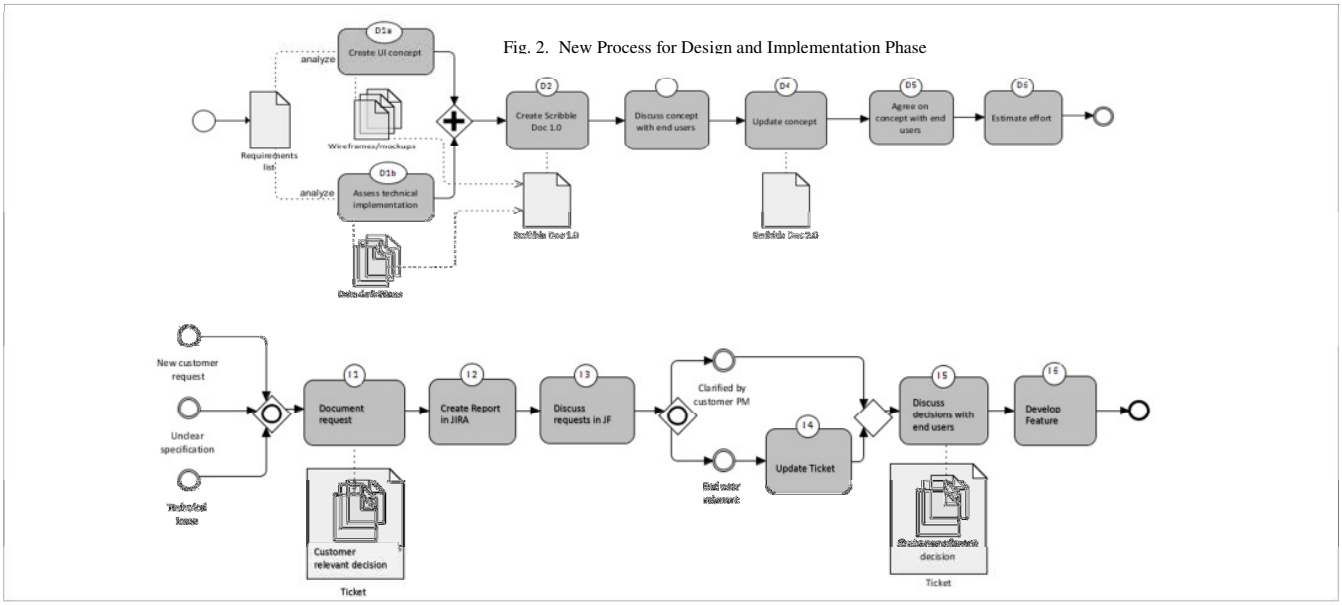

Fig. 2. New Process for Design and Implementation Phase

repositories (scribble doc and the JIRA tool) can be used to circulate information between all users.

An overview of the instantiation decisions is given in Table I. The descriptions of the newly designed overall processes based on the instantiated method are described in the following paragraphs for the design and implementation phase (see Figure 2).

Design phase. The new process at the beginning of the design phase starts when the IT personnel receive the requirements list of the customer PM. The first step (D1a) is to create a user interface (UI) conception, which shows how the new requirement will be included in the existing application. These decisions are captured in mockups. In parallel there is step D1b, in which the developers assess the technical implementation of the feature. The results of the assessment are captured in data definitions.

The following step D2 is to combine the mockups with data description in the scribble doc. In step D3 the first version of the scribble doc is discussed with the 28 user representatives. Within this first workshop all requirements/features are presented through mockups the data definition. There is a direct discussion between the IT personnel and the users to align the current understanding of the rationale for each feature. In step D4 the scribble doc is updated with the new information gained in the workshop. The result is the scribble doc 2.0. After the new description of all features, there is a second workshop with the 28 user representatives to agree on that scribble docand sign off the scope of this release. After the sign off, the estimate for effort is fixed.

Implementation phase. During development three different user-related events can occur: a new customer request from the customer PM is issued, an unclear specification requires a decision, or a technical issue requires information from the customer. After one of these three events occurs, the request is documented in the new field "customer-relevant decision" within a JIRA ticket (step I1). The documentation is structured into questions, alternatives and implications of the decision, as suggested in our previous work [4]. The documentation within the ticket has two advantages. Firstly the IT personnel, who receives the request, has to put it into context. Secondly, the documentation with alternatives and implications requires thinking about possible solutions and consequences. Step I2 comprises the preparation of the weekly jour fixe with the customer PM. For that, the IT PM creates a report of all open entries within the field customer relevant decisions of the project. As a first level, all requests from the report are discussed in the jour fixe with the customer PM and classified with the IT PM, the designer and the developers. In the discussion it is decided, whether the request can be clarified by the customer PM or should be discussed in the second level with the user (step I3). Requests that are clarified by customer PM, will be directly updated in JIRA. All user-relevant decisions are collected. If there are about five user-relevant decisions or if a defined period of time has passed (e.g., four weeks), a workshop with four to five selected user representatives takes place (step I4). The IT personnel presents the required decisions in the format question, alternatives, and implications. They highlight the affected feature visually. In a joint discussion of the IT personnel with the users an alternative is selected. In step I5 the field customer-relevant decision is updated in JIRA with the previously made decision. The last step I6, comprises the development of the feature.
In order to describe the instantiation as detailed as possible we extracted an example for each process.

Example Design Phase – Inactive employees. In the current iPeople solution, only active employees are displayed in the organizational tiles. The example deals with the extension to also display inactive employees. The initial requirement from the first document is:

As-is state: Currently only employees with the employment state 3 = active are displayed in iPeople.

Target state: In future all employees that have a data record in the table IT9006 should be displayed in the iPeople solution

It is obvious that this description is on one hand very specific, as a source (data base table IT9006) is given, on the other hand there is no indication why this information is needed. This makes it hard for designers or developers to include that feature within the application. Based on this initial requirement, two steps occur in parallel.

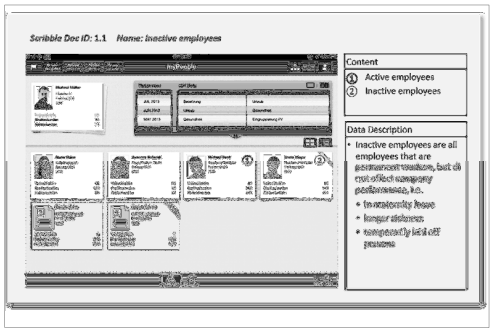

Fig. 3. Scribble Doc 1.0 of Inactive Employees

The designer creates the mockups (D1a), which is displayed in the screenshot is Figure 4. The designer decides to include a banderole indicating, whether an employee is inactive ("abwesend"). Furthermore, the designer describes the content on the screen, i.e. active and inactive employees. In parallel, the developers describe the data description. As there is no information within the initial requirement, the developer analyzes the data table IT9006, and documents the results in the data description (D1b): "Inactive employees are all employees that are permanent workers, but do not effect company performance, i.e. in maternity leave, longer sickness, or temporarily laid off persons". This initial scribble doc 1.0 (D2), consisting of one mockup and a data definition for each feature, is then discussed with the 28 user representatives in a workshop (D3). Thus, the IT personnel presents the question in the workshop: "why do you need to display inactive employees? " The user might explain that they want to have this feature, as they want to replace a currently used paper-based list (cost center list). To ensure they can abandon that list, they need to get an overview of all outstanding cost. In the direct discussion, it is revealed that not only inactive employees, but also laid-off people could still have outstanding bonuses or travel cost. The mockup is updated with a third category "laid-off employees", which also get a banderole ("entlassen"). Furthermore, it is important that for this category no pictures of persons are shown, as the data privacy protection requires this. In addition, the data description is adapted and the changes are highlighted (e.g. bold, italic) (D4). This new version of the scribble doc is then presented to the 28 user representatives. When all agree on the mockups (D5), the design decisions are final and the effort for implementation can be estimated (D6).

Example Implementation Phase – Working time report

The example for the implementation phase is the feature: working time report. This feature had six different specifications in the actual development; the last change was specified even after development closed. Therefore, it is clear that the specification has not been detailed enough. The requirement is to include KPIs about breaches of working time standards (e.g. unused breaks, overtime, etc.) in the detailed view of an employee. For example there should be a KPI to describe deviation of actual vs. allowed working time. Three different events can cause decisions in the implementation phase. First, the customer can request a new feature or an adaptation of a feature. An example of such a request is that the thresholds for the breaches should not be fixed values (e.g. show breach, if more than 25% of assigned employees have illegitimate overtime), but rather be customizable for each country. Secondly, a technical issue can occur during development, for example it was not clear to the back end developer, where the mapping of actual breaches to possible breaches is defined in the back end system. Thirdly, ambiguities in the specification might come up. For example it was not explicitly specified what should be displayed in case there is a breach in the working time report.

The first step of the process (I1), in the case of these three events is to document the request within the new field in the JIRA ticket of the affected feature. For the example of an unclear specification event, the question is "what should be displayed, if there is a breach in the working time report?" There are three alternatives, the first one, the easiest from an implementation point of view, is to just display 0%. However, that might be confusing for users as the standard threshold means actually fewer than 25 %. The other two alternatives (show "-" or "n/a") both require more implementation effort, as the front end needs to be adapted, but it might be easier to comprehend for the users. The next step (I2) is to build a report in JIRA about all open customer relevant decisions. The clear advantage of using a separate field within tickets is that it is possible to filter against that field. Within the jour fixe the customer relevant decisions are discussed with the customer PM (step I3). The discussion of the new customer request reveals that it is a change request, but it is not too much effort. Therefore it can be implemented within the current release. The discussion about the technical issue shows that the customer PM needs to ask the customer IT department, where those mappings are available in the back end system. Thus, those two decisions are classified as "clarified by customer PM". However, the last decision about the unclear specification needs input of the users. The next step I4 is to discuss the user-relevant decisions with four to five selected user representatives. Within the workshop the IT personnel explains the open question with alternatives and implications. In order to ensure that the user representatives understand the context of the question, the scribble doc ID, as well as a visual mockup is shown. The discussion with the users leads to the decision to use alternative 2. Nevertheless it is important that the users also understand the consequences of higher implementation cost. After the decisions are made, the JIRA ticket is updated to ensure the traceability of the decision (step I5). When the result of the decision process is documented, the implementation of the feature can continue.

As we only simulated this instantiation, we wanted to get feedback to this instantiation of the UDC-LSI method from the project participants. This evaluation is described in the next section.

IV. Evaluation of the UDC-LSI Method in practice

The general goal is to evaluate this instantiation of the UDC-LSI method in the unit of analysis (iPeople project) from the perspective of project participants. In particular, we wanted to understand the feasibility, effectiveness, efficiency, and acceptance of the instantiation of the UDC-LSI method.

Feasibility (RQ 1). We analyzed the feasibility within three dimensions: internal development process, for this system, with this customer. All participants believe it is feasible to perform this instantiation. However a small part thinks it requires high effort, especially for communication in the design phase. In total, seven participants (78%) consider "Availability and organization of 28 user representative from different countries is complex". The steps in the process that concern the documentation are considered easier than the steps that concern communication. 56% of participants reasoned that as "mockup usually exists, so extension to scribble doc is small". The process suggested for the implementation phase is rated with lower effort than process for design phase. This can be reasoned through the existence of the JIRA tool (named by 33% of participants). Still, there were three medium effort ratings, all from back end developers, since it requires a lot of writing. Overall, the participants think this process has low to medium effort, since the process is proven in other projects (named by 44% of participants). The instantiation of the UDC-LSI method is also considered feasible from the system dimension, the majority (69%) even thinks it can be performed well. The results also show that the new process is harder to implement in the design phase than in the implementation phase. In total the results show a higher rating than from the process perspective. This can be explained, since the system itself does not have such a high influence on the UDC-LSI method (named by 44% of participants). The third category, whether it is feasible to perform the UDC-LSI method with this customer, has the lowest rating regarding feasibility. However, there is still a majority of 47%, which believes it can be performed. But especially the discussions with the 28 user representatives are considered by the majority (67% of participants) to be very hard to conduct with this customer. The only answer of "not at all" is from the IT PM, as her job would be it to convince customer PM. Altogether we consider H1 is confirmed.

Effectiveness (RQ 2). For effectiveness, we mainly focus on system success aspects (i.e. user satisfaction, ease of use, system use, project's time and budget, system quality, and data quality). We asked the project participants, whether they believe that system success would increase, stay unaffected, or decrease with the performance of the UDC-LSI method in the iPeople project. The results show that the majority of answers (69%) indicate an increase of system success aspects. About 30% specify that the aspects are unaffected and only one answer (1%) shows a low decrease. For the aspect user satisfaction all participants believe that it will increase, due to that fact that users get more understanding and feel integrated (named by 67% of participants). A majority (67%) believes ease of use will increase, however three project participants (33%) believe it is unaffected, as this is the job of the designer, independent of measurements concerning UDC. For system use the majority (56%) believes in an increase. But 44% believe it is unaffected, as the system usage of the business application is mandatory for the users. The vast majority of 89% believes that project's time and budget will improve due to performance of the UDC-LSI method, as the clearer scope will lead to better planning (named by 67% of participants). However, one participant believes that the effort from a timing perspective is so high through the organization of the workshops, that project's time and budget will slightly get worse. For system quality 56% of the participants believe in an increase. However, 44% believe this aspect is already quite high and cannot be influenced by the suggested UDC-LSI method instantiation, but only through refactoring of the code. The last aspect data quality is considered to stay unaffected by the majority (54%). The rationale of five participants (56%) is that the iPeople system only reads data from an existing backbone, thus changes in the iPeople process would not affect the data. Nevertheless, 44% believe it can be increased as a better understanding of the scope will ensure better usage of the data. For some aspects the opinions are spread between unaffected and increase, however for five out of six aspects the majority expects an increase. Therefore H2 is confirmed. This means the UDC-LSI method has a positive effect on system success.

Efficiency (RQ 3). Regarding efficiency, we asked the project participants for their opinion on the effort-benefit ratio of the four parts of the instantiation of the UDC-LSI method. The vast majority (81%) of the project participants agrees or strongly agrees that the benefits of executing the instantiation of the UDC-LSI method compensate its effort. The agreement is a bit higher for the documentation parts (89% agree + strongly agree) than the communication parts (72% agree + strongly agree). Between the two phases it is almost identical (design phase 83% agree + strongly agree and implementation phase 78% agree + strongly agree), with a slightly higher agreement wrt the design phase. There are only two votes that disagree or rather disagree. Both are from the IT project manager, who believes that the effort to get user representatives is too high and thus does not compensate for the effort. Since the vast majority agrees that for all parts of the UDC-LSI method the benefits outbalance the effort, hypotheses H3 is confirmed.

Acceptance (RQ 4), Regarding acceptance, we build upon the Technology Acceptance Model (TAM) [5]. Therefore we checked for perceived ease of use, perceived usefulness, attitude towards using and the behavioral intention to use. According to TAM an agreement towards those criteria forecast actual system use, thus acceptance. Table II shows the results for acceptance. We also aggregated the votes, with negative ((strongly|rather) disagree), neutral, and positive ((strongly|rather) disagree). The majority of project participants agree (97%) that all parts of the UDC-LSI method are easy to understand and easy to use. The only two negative answers with the communication part of design and the implementation phase are from the IT project manager, who thinks that persuading the customer PM to have discussions "only" in the jour fixe and then with users will not be possible in practice. The majority of project participants agree (97%) that all parts of the UDC-LSI method are useful. The only negative answer is from a back end developer who thinks a more detailed

description of the scribble doc would be required to actually explain user requirements. All project participants have a positive attitude towards usage (100%) Since the project participants agree to the perceived ease of use, have a positive attitude towards using and a behavioral intention to use, H4 is confirmed.

Overall, the results showed a clear positive assessment of utility (i.e. feasibility, effectiveness, and efficiency) and usability (i.e. acceptance by the future users) of the UDC-LSI method in the opinion of the project participants.

TABLE II. ACCEPTANCE OF THE UDC-LSI METHOD

TAM Element	Phase	Strongly disagree	Disagree	Rather Disagree	Neutral	Rather	Agree	Strongly Agree
Perceived Ease of Use	No. of votes	0	1	1	0	3	20	47
	Percent	0%	1%	1%	0%	4%	28%	65%
	Aggregated votes	2			0	70		
	Aggregated percent	3%			0%	97%		
Perceived Usefulness	No of votes	0	0	1	3	9	13	64
	Percent	0%	0%	1%	3%	10%	14%	71%
	Aggregated votes	1			3	86		
	Aggregated percent	1%			3%	96%		
Attitude towards Using	No of votes	0	0	0	0	1	9	26
	Percent	0%	0%	0%	0%	3%	25%	72%
	Aggregated votes	0			0	36		
	Aggregated percent	0%			0%	100%		
Behavioral Intention to Use	No of votes	0	0	2	4	3	13	50
	Percent	0%	0%	3%	6%	4%	18%	69%
	Aggregated votes	2			4	66		
	Aggregated percent	3%			6%	92%		

V. THREATS TO VALIDITY

In the following we consider threats to validity based on [3].

Construct Validity: The case study design included a plan on how the data of the different sources is used to answer the research questions. This helps to ensure that the feedback of project participants reflects their true opinion [6]. Also, we ensured construct validity through data source triangulation. We obtained data from different roles with different backgrounds and experiences levels in project to ensure a holistic view. For the instantiation the researcher applied the method to ensure correct and complete instantiation. In the evaluation, there is a threat that interview questions could be interpreted differently by researcher and interviewee. But we explicitly presented definitions and personal interviews enabled questions by the interviewee in case of a lack of clarity. The questionnaire and the used presentation were checked for understandability by several researchers. All interviews we recorded with the consent of the interviewees which enabled us to transcribe all open questions.

Internal Validity: A potential threat is that the project participants were biased towards acceptance of the method as they were only presented with a hypothetical instantiation. But, we explicitly adverted in the interviews that they should assess the method objectively.

External validity: A possible threat in the evaluation part is that we could only interview project participants from the IT team, we therefore missed the perspective of users. We mitigated this by including roles that have a similar background as possible users.

Reliability: All assessments and interviews were done by one researcher. On one hand this ensured consistency [3], on the other hand another researcher could interpret the data differently. During design, data collection and analysis the researcher continuously documented every step that was done. Each step got peer reviewed by a second researcher.

VI. CONCLUSION AND OUTLOOK

In this paper, we report on a case study on utility and usability of the simulated UDC-LSI method in the real-world iPeople project. The simulated instantiation was presented and evaluated with regards to feasibility, effectiveness and efficiency. Furthermore we evaluated the acceptance of the method. During the simulated instantiation, we analyzed the four parts of the UDC-LSI method. An interesting insight is that we needed *to define two different processes for the design and implementation phase*, since they differ in the nature of decisions, documentation and tool. The evaluation showed a *clear positive assessment of utility (i.e. feasibility, effectiveness, and efficiency) and usability (i.e. acceptance) of the UDC-LSI method.* The *feasibility is considered higher for the documentation part than the communication part*. This makes sense, since the documentation part can be done internally within the IT Company, whereas the communication part requires contact with the customer. An *open issue is how the communication with user representative should look like.* Especially, how many representatives are required and what format (i.e. personal or online meeting) works best. This could not be finally answered within this case study. Since we did not have the chance for discussions with users, this is up for future work. Furthermore, to complete the evaluation of our method a real life usage of the method in a ongoing project is an important step for future work.

ACKNOWLEDGMENT

Gunter, Horst, Lydia, Manuel, Maria, Michael, Neha, Nico, and Sven as well as all employees of the sovanta AG for their time and support during the case study.

REFERENCES

[1] A. R. Hevner, S. T. March, J. Park, and S. Ram, "DESIGN SCIENCE IN INFORMATION SYSTEMS RESEARCH," *MIS Q.*, vol. 28, no. 1, pp. 75–105, 2004.

[2] U. Abelein and B. Paech, "Understanding the Influence of User Participation and Involvement on System Success - a Systematic Mapping Study," *Empir. Softw. Eng.*, no. Iivari 2004, p. in press, Dec. 2013.

[3] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empir. Softw. Eng.*, vol. 14, no. 2, pp. 131–164, Dec. 2008.

[4] U. Abelein and B. Paech, "A Descriptive Classification for End User - Relevant Decisions of Large-Scale IT Projects," in *Cooperative and Human Aspects of Software Engineering (CHASE), 2013 6th International Workshop on*, 2013, p. accepted.

[5] F. Davis, R. Bagozzi, and P. Warshaw, "User Acceptance of Computer Technology: A Comparison of two Theoretical Models," *Mangagement Sci.*, vol. 35, no. 8, pp. 982 – 1002, 1989.

[6] R. Wieringa, "Writing a Report About Design Research," no. February, pp. 1–9, 2007.