IN SEARCH OF A FRAMEWORK FOR USER-ORIENTED DATA-DRIVEN DEVELOPMENT OF INFORMATION SYSTEMS

TEA MIJAČ[1]
MARIO JADRIĆ[2]
MAJA ĆUKUŠIĆ[3]

ABSTRACT: *Although iterative and continuous improvements are common in the design and development of contemporary information systems, some interesting research directions are arising from general trends towards user-oriented data-driven approaches. Since this is a relatively active research area, the paper aims to provide a brief overview of the main concepts and research directions and to define the setup and the context for the proposed user-oriented data-driven information systems development framework. Experts from the field of information systems are interviewed and their feedback is used to evaluate different aspects of the proposed framework.*

Key words: *user-oriented development, data-driven development, framework, user experience*

JEL classification: M15

DOI: 10.15458/ebr.89

1 INTRODUCTION

End-user requirements are a starting point for any information systems (IS) development, and as such, they are regarded as a critical success factor (Maguire, 2013; Medeiros et al., 2015). The need for the end-user involvement in the IS development process has been reported since 1982, although this involvement was far more elementary at the time (Robey & Farrow, 1982). The studies often confirm that the main challenges in the IS development are insufficient user involvement and the constant change in user requirements (Medeiros et al., 2015). One recent study indicated that changing user demands can cause 70% of the problems in the process (Geogy & Dharani, 2016). Some of the main causes for failures of software development projects are incorrect identification of the requirements, changes in user requirements, as well as their uncontrolled evolution during the project development lifecycle. Changes in the requirements impact the development process negatively, potentially causing budget breaks and, in the end, the result does not meet all customer

1 University of Split, Faculty of Economics, Business and Tourism, Split, Croatia, e-mail: tea.mijac@efst.hr

2 University of Split, Faculty of Economics, Business and Tourism, Split, Croatia, e-mail: mario.jadric @efst.hr

3 University of Split, Faculty of Economics, Business and Tourism, Split, Croatia, e-mail: maja.cukusic @efst.hr

needs (Garcia & Paiva, 2016). For example, the studies have shown that 60% to 80% of IT projects in the public sector fail in that regard (Brown, Fishenden, and Thompson, 2014).

For this reason, many researchers point out that Requirement Engineering (RE) is the earliest and most crucial phase of IS development (Amokrane et al., 2015). The purpose of RE is to understand the problem that arises from the transformation of different needs of all the stakeholders into the structured requirements that form the ultimate solution, taking into account all the requirements that could change over the entire life cycle (Wiesner, Nilsson, and Thoben, 2017). The main challenge is to "translate" users' requests into what the system ultimately provides (Geogy & Dharani, 2016). Managing requirements is very important because it is necessary to keep track of any changes that occur in user requirements and to ensure that the adequate changes are made to meet the needs of all stakeholders (Inayat et al., 2015; Geogy & Dharani, 2016).

Due to the extremely important role that user requirements have in the process of development or information system re(design), there is a large number of research studies regarding the different degrees of formalities for these requirements (Bjarnason et al., 2016). User requirements can refer to almost all major features, usability, accessibility, navigation, electronic resource formats, and more, and can be divided into functional and non-functional requirements (Medeiros et al., 2015).

In focusing on the user and his/her needs, it is necessary to set up a development process that incorporates the feedback loop, so that the result could be a product that provides the usability and other success factors for end-users for a longer time (Richre & Fluckiger, 2014). User-oriented development as a paradigm is applicable in all areas where users interact, "communicate" with the software and intensively use it. The main advantage of user-oriented development is the idea that the testing and optimization of the end-product is done in co-operation with end-users in all phases. This approach enables some of the "hidden" demands to be brought to light. What is more, the information on users' perspectives as well as weak points must be one of the basic inputs for the improvement of business processes within organizations (Vanwersch et al., 2015). Availability of user data has led to the evolution from the user-oriented development towards data-driven development. Authors anticipate a shift to a data-oriented and user-oriented paradigm, especially when it comes to identifying, collecting, and managing user requirements for development (Maalej et al., 2015). Business analysts at times do not have enough information about end-users, so a lot of aspects of the requirements remain incomplete (Maguire & Redman, 2006). New technologies make it easier to understand user needs, as well as to provide a much better quality of interaction – this leading to a more successful collection and analysis of user needs as a basis for (re)designing and developing user-oriented IS. Data-driven development assumes that the data collected informs some future decisions, i.e., influences the decision-making process (King, Churchill, and Tan, 2017). The development team must take into account all the available data when deciding about the development of the (next) version of the IS (Maalej et al., 2015). The data-driven approach contributes to an easier understanding of user needs, while agile approaches to development enable rapid response to changed requirements (Brown, Fishenden, and Thompson, 2014).

In general, data-driven decision making is related to higher productivity, higher return on assets, equity, and market value (Brynjolfsson, Hitt, and Kim, 2011). In that sense, data can help in measuring the effect of design on users, and business results (King, Churchill, and Tan, 2017). Software developing companies thus need to focus on exploiting data to gain a competitive advantage as there is unfulfilled potential for data-driven approach in agile software development context (Svensson, Feldt, and Torkar, 2019).

Motivated by the presented argumentation, this paper provides an overview of the most important studies and fundamental concepts underlying contemporary user-oriented data-driven IS development in Section 2. In an attempt to structure and map out the potential research direction in this specific area, a framework is sketched and described in Section 3. It provides a basis and a good starting point for interviews with experts. The following research questions have been derived: How do experts understand and acknowledge the importance of data-driven approaches in IS development?; What is their personal experience in using related approaches?; What are the potential advantages and obstacles to following the proposed user-oriented data-driven IS development approach?; and What is their overall opinion about the proposed framework? Expert feedback is presented in Section 3, followed by conclusions in Section 4.

2 THEORETICAL BACKGROUND

2.1 The importance of User eXperience (UX) in information systems development

User eXperience (UX) is a multidimensional construct that includes different user reactions to the product throughout the entire user journey (Lemon & Verhoef, 2016). Each user has a unique set of preferences, knowledge, characteristics, as well as limitations that potentially affect UX and make them extremely subjective and depend on the context (Cormier & Lewis, 2015; Halvorsrud, Kvale, and Følstad, 2016). Although the phenomenon of UX is widely accepted in the industry, there is still no agreement in the scientific community about the comprehensive theoretical UX model (Zarour & Alharbi, 2017). The quality of UX is measured, for example, by usability observed when the user interacts with the system (Sohaib & Khan, 2010). Standard ISO 9241-11 defines usability as a "measure in which specific users can use a particular product to achieve certain goals, effectively, efficiently and with pleasure, depending on the context in which they are used." Studies have shown that end-users prefer a website with a higher usability rating, although one should bear in mind that the user requirements regarding usability and design depend on the type of digital service and users themselves (Ilbahar & Cebi, 2017). The results of the studies also confirm that user satisfaction has a positive impact on the user's intention to use the site but also on the use in general and that the perceived usefulness of the site has a positive impact on the customer's satisfaction (Cohen, 2006; Belanche, Casaló, and

Guinalíu, 2012). Usability is believed to be one of the main factors affecting the level of use of digital services in the public sector (Huang & Benyoucef, 2014) as well. Satisfaction with the use of digital public services is the ability of users to find the information they need and to experience a service that addresses their problems (Reddick & Roy, 2013).

Concepts of UX and an agile approach in development are compatible (Sohaib & Khan, 2010). The agile approach allows for easier adaptation when changes occur, either in the prioritization of the requests or changes in the content itself (Heikkila et al., 2015). It is an important advantage since the consensus is that most of the problems in the development are caused by overly frequent changes in user requirements and a low level of end-user involvement (Medeiros et al., 2015).

2.2 Data-driven approach in information systems development

In the context of IS development, it is important to emphasize the emergence of big data as one of the progressive technological areas, which enables the availability and analysis of extremely large amounts of data. Every day we witness the production of a growing number of digital artifacts (Tomitsch, 2018). The value of big data is not its size – it is that it can offer new kinds of information to study – information that has never previously been collected (Stephens-Davidowitz, 2017). The amount of data produced by users nowadays exponentially increases, and by reviewing the data, generated by users while interacting with IS, development teams could understand better what users are really doing and how they respond; and in addition, some defects are detected in real-time (Spiess et al., 2014; King, Churchill, and Tan, 2017). Development teams in the IT sector already collect huge amounts of implicit feedback in the form of usage data, error records, and sensor data (Maalej et al., 2015). Since the user habits and interests are changing rapidly, and new trends emerge daily, while having in mind that UX is ultimately subjectively, dynamically and contextually dependent (Halvorsrud, Kvale and Følstad, 2016), designers and developers have no choice but to take into consideration the data generated by different user actions and feedback collected from the overall user journey (Lemon & Verhoef, 2016).

Authors point out that tracking the objective user data is extremely important and that this data is a key component when evaluating the UX (Sengers et al., 2008). The problem is that user data has been so far often interpreted on intuition rather than actual data (Havice, 2017). Direct contact with users is the key to achieving user-oriented approaches and fulfilling their expectations (Kujala, Kauppinen, and Rekola, 2001). One can claim that there is nothing more direct than the data that users produce themselves. People produce data, so it is both important to enable its collection and to provide opportunities for end-user feedback (Tomitsch, 2018). Users should be allowed to provide the feedback while using the system, and whatever comes out of the exercise should be the best solution (King, Churchill, and Tan, 2017).

The more data is available, the greater the chances are of understanding the user and his behavior (Spiess et al., 2014; Anderson, 2015). When tracking objective user data, some authors distinguish between active and passive monitoring, i.e., there are two types of data: implicit and explicit (Maalej et al., 2015; Liikkanen, 2016). Passive tracking includes tools that capture users' actions and features, while active tracking consists of asking feedback about the service, directly (through polls) or indirectly (chat conversations). In other words, specific, explicit data can also come from the comments of the users themselves even if self-initiated (Maalej et al., 2015). The tools used for passive tracking usually address the challenges of automatic collection and analysis, and authors label that type of data as implicit data (Maalej et al., 2015; Liikkanen, 2016). An example of implicit feedback is a large concentration of clicks on one location that can point to a potential problem.

Authors define two categories of metrics used in the context of a data-driven approach to IS development (Rodden, Hutchinson, and Fu, 2010):

- PULSE metrics: number of page views, earnings, activities, and similar.
- HEART metrics: (1) Happiness – aesthetics, and perceived ease of use, (2) Engagement – frequency of use and / or intensity, (3) Adoption – number of unique users in a given period, (4) Retention – how many of the users from a given time period are still present in some later time (5) Task success – could be measured with effectiveness, efficiency and error rate.

Diverse data is not only generated internally within software-intensive companies but also from different sources (Svensson, Feldt, and Torkar, 2019); for example, categories of data types are:

- Logs – can reflect relevant information about the past and current state of the system and could be of different types depending on predefined properties.
- Visitor metrics – referring to user data collected through external tools, such as Google Analytics, HotJar, and similar.
- Visual metrics – for example, heatmaps and clickmaps.

Table 1 itemizes the categories mentioned above and sources while providing references to authors that studied the availability and importance of specific data for various aspects of the IS success.

Table 1: *Common objective data sources and types used in data-driven IS development*

Categories of data types	Objective data types and sources	Authors
Logs	Logs in general	Rodriguez, 2002; Rodden, Hutchinson. and Fu, 2010; Andrica & Candea, 2011; Inversini, Cantoni, and Bolchini, 2011; Gordillo et al., 2014; Harrati et al., 2015; Maalej et al., 2015; Garrido et al., 2017; Grigera et al., 2017
	Keyboard	Saadawi et al., 2005; Andrica & Candea, 2011; Espada, Garcia-Diaz, and Crespo, 2012
	Usage patterns	Harrati et al., 2015
	Task duration	Oertel & Hein, 2003; Saadawi et al., 2005; Harrati et al., 2015
	Time metrics	Grigera et al., 2017
	Mouse clicks	Oertel & Hein, 2003; Andrica & Candea, 2011; Harrati et al., 2015; Garcia & Paiva, 2016; Frantz, 2018
	Task success	Saadawi et al., 2005; Rodden, Hutchinson, and Fu, 2010; Inversini, Cantoni, and Bolchini, 2011
	Scrolling	Au et al., 2008
	Error rate	Au et al., 2008; Rodden, Hutchinson, and Fu, 2010
Visitor metrics	Organic clicks	Rodden, Hutchinson, and Fu, 2010; Bakaev, Bakaev, and Mamysheva, 2016; Lee et al., 2016
	Bounce rate	Bakaev, Bakaev, and Mamysheva, 2016; Lee et al., 2016
	Traffic; the number of unique visits	Rodden, Hutchinson, and Fu, 2010; Bakaev, Bakaev, and Mamysheva, 2016; Lee et al., 2016
	User activity	Rodden, Hutchinson, and Fu, 2010; Bakaev, Bakaev, and Mamysheva, 2016; Lee et al., 2016
Visual metrics	Session recording, video recording	Oertel & Hein, 2003; Maalej et al., 2015
	Eye movement tracking	Oertel & Hein, 2003
	Heat maps	Gordillo et al., 2014; Courtemanche et al., 2017; Grigera et al., 2017

The importance of using objective data in the context of improving user experience and IS development has been acknowledged in the literature. However, studies on how practitioners and IS developers use objective data as a part of their data-driven strategies and IS-related decisions are scarce (Svensson, Feldt, and Torkar, 2019) and still focus on collecting objective data using Google Analytics almost exclusively. Examples include improving the user experience while anticipating potential customer complaints by telecom users (Bao, Wu, and Liu, 2017) or in online library users (Lee et al., 2016). The latter study also used A/B testing for collecting and using data during the design stage to get feedback in the early stages of development (Lee et al., 2016). It is essentially an experiment used when changes are made to a product/service to measure the effect (Rodden, Hutchinson, and Fu, 2010; Lee et al., 2016; King, Churchill, and Tan, 2017). Controlled A/B tests are tests where two similar populations of users interact with different artifacts, and their responses can be rigorously measured and compared (Rodden, Hutchinson, and Fu, 2010). It is possible to automatically detect some issues without being a usability expert (Grigera et al., 2017), as exemplified by a recent study of differences between two different versions of a webshop to scope a few improvement actions (Mijač, Jadrić, and Ćukušić, 2018).

Authors agree that this type of experimentation is a well-established practice in online systems, yet it has not been applied extensively in large-scale development of embedded software systems (Olsson & Bosch, 2014). Olsson and Bosh (2014) proposed a model in which the final step is where the entire R&D organization acts based on customer data, and where the deployment of software functionality is seen as a way of evaluating what the customer needs. Their HYPEX model (Hypothesis Experiment Data-Driven Development) supports companies in running feature experiments to shorten customer feedback loops. They use strategic product goal to generate feature backlog and later analyze the gap between expected and actual behavior.

Software development companies are increasingly aiming to become data-driven by trying to experiment with the products used by their customers continuously. Controlled experimentation is becoming the norm in the software industry for reliably evaluating ideas with customers and correctly prioritizing product development activities (Fabijan et al., 2017).

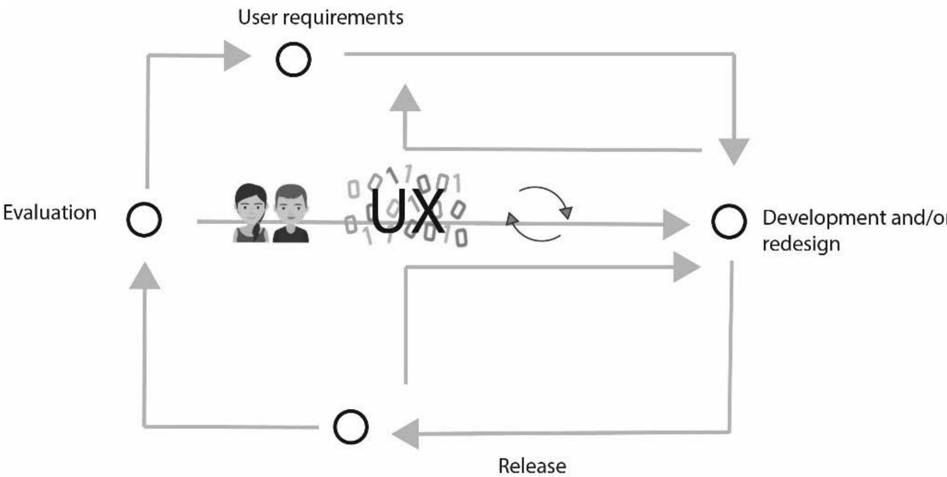
A survey conducted in Sweden shows that there is a lot of potential for data-driven decision making in agile software development but currently unfulfilled (Svensson, Feldt, and Torkar, 2019). The development team should be able to incorporate the demands of all users when deciding what to develop and when to release the application (Spiess et al., 2014; Maalej et al., 2015). Collecting data, learning from it, and making iterations in the (re)design phase to find positive and negative elements leads to data-based approaches that provide systematic observation and a more ambitious approach for modeling and development (King, Churchill, and Tan, 2017). Organizations that are developing information systems are faced with the difficult choice of picking the right Software Development Life Cycle (SDLC) (Balaji & Sundararajan Murugaiyan, 2012). SDLC adheres to important phases that are essential for developers, such as planning, analysis, design, and implementation and evaluation. Traditional development is an approach characterized by slow cycles with ad-hoc customer feedback processes, and where the customer is not well integrated into the product development process (Olsson & Bosch, 2014), an issue solved by modern agile development approaches.

2.3 Formulating a user-oriented data-driven information systems development framework

Following the presentation of important aspects for the IS development and the potential of available objective (user-generated) data, the general concept of a framework for user-oriented data-driven IS development is proposed. It draws on an iterative approach to system design, similarly to Boehm's spiral model of IS development – a combination of iterative and linear software development approach (Boehm, 1988). The spiral model has two main features: (1) a cyclic/iterative approach that serves the gradual development of a system, and (2) reducing the degree of risk (Barry & Wilfred, 2000). As it is a layered development, developers go through all the stages of development several times. Agile methodologies already mentioned above are based on the principles of continuous improvement and testing based on feedback. The agile approach is user-oriented, unlike the process-oriented spiral model. In the spiral model, full specifications are known in advance, but the development is done in layers.

The first concept of the framework for user-oriented data-driven information systems development is sketched in Figure 1. It encapsulates the main phases already well known in the process of developing the information system. The first phase is named "User requirements" – which stands for the initial phase including planning, analyzing and collecting the user requirements. After collecting all the necessary input, the next phase begins – "Development and/or redesign". Here, term redesign has been added to put an additional highlight on the redesigning process as many systems running today need frequent rework and repositioning towards the user's changing preferences. It is also a typical phase in system development, and one can argue that it represents the core phase. The loop represents a repetitive action until all user requirements are collected. It is in line with the user-oriented paradigm argued in detail above. After the development of an information system, it is ready for its first "Release" (implementation). Only after releasing the initial version of the software, it is possible to continue with the "Evaluation" phase – which should represent the last phase of this process. Since the whole idea is based on the agile approach, delivering working software in short iterations has to be a condition, with teams that are frequently involved in short-term decisions and need to adapt to a fast decision-making process (Svensson, Feldt, and Torkar, 2019). For that reason, the loops between evaluation and development (redesign) have been added/envisaged, since the changes in user requirements need to be incorporated in subsequent versions. This relationship contains a dimension which, according to the theoretical background, stands as a crucial element for the user's intention to use the system and also has a positive impact on the user's satisfaction. Precisely this dimension represents an opportunity to incorporate data-driven approach, so apart from (or even instead of) asking the users whether they are satisfied with the proposed/existing solution, additional objective (passive) data can be used for enriching their feedback. As mentioned before, software development is a cyclic process, so after the "Evaluation" phase, the proposed process does not finish but goes back to the beginning. The sketched concept of the framework was explained and discussed thoroughly with experts who helped to formulate the amended version of the framework that is presented and explained further in the paper.

Figure 1: *The first concept of the framework for user-oriented data-driven information systems development*



Source: Own elaboration.

3 EXPERT EVALUATION OF THE PROPOSED FRAMEWORK

3.1 Methodology and description of the procedure

The qualitative research method was used to explore our understanding of data-driven and user-oriented paradigm in IS development and to get feedback on the general idea of the presented framework. Also, the intent was to explore and discuss how software organizations may use this framework in the future in an agile software development environment. There are numerous advantages and recommendations for using the qualitative methodology in the field of IS (Marshall et al., 2013). It is especially relevant taking into account the purpose of this exploratory research study – to determine whether the proposed framework could be used in practice and, in general, to find out the pros and cons of using such data-driven approach in IS development. Consequently, an in-depth semi-structured interview was identified as a suitable data collection method for this specific purpose. Semi-structured interviews generally last for a considerable amount of time depending on the particular topic (Smith & Osborn, 2007) and there is no rule on how many participants should be included (Pietkiewicz & Smith, 2012). Sample sizes in qualitative research customary involve a range of 3 to 10 individuals (Creswell & Creswell, 2018).

Semi-structured interviews with four experts were conducted and recorded over six months (two in November 2018 and two in April 2019). In line with the qualitative approach methodology, the interviews lasted as long as the comments started to repeat – the duration of each interview was approximately one hour, in a total over four hours. All the interviews were conducted in the Croatian language and recorded with the permission of the participants. Basic sociodemographic information about the interviewees is presented in Table 2, while other job-related and topic-related characteristics are presented in the following section in relation to their feedback.

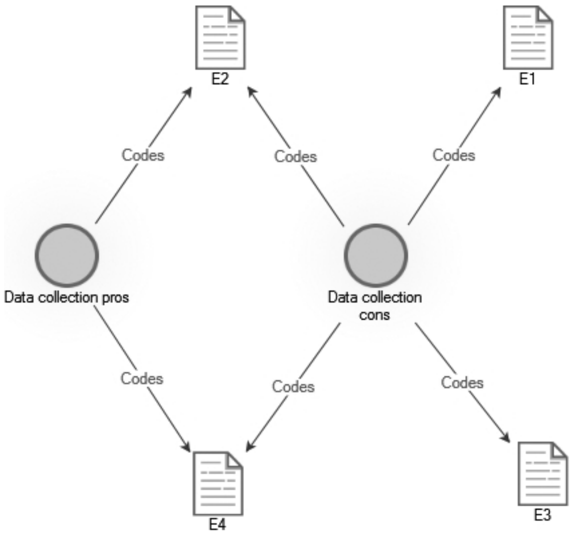
Table 2: *General characteristics of the four interviewed experts*

	Expert 1	**Expert 2**	**Expert 3**	**Expert 4**
Age	31	30	31	35
Gender	Male	Male	Female	Male
Education	Master's degree	Master's degree	Master's degree	Master's degree
Income (in HRK)	15,000 – 20,000	10,000 – 15,000	Less than 10,000	15,000 – 20,000

The interviews were based on open-ended questions to ensure that the same basic topics were covered with all the participants, but the format was kept flexible to allow the conversation to develop naturally and to elicit views and opinions. The rationale was that they would reveal a lot more about relevance and maxims connected with their positions and functions: they carry on talking about their activities, give examples, or use other forms of elaboration (Bogner, Littig, and Menz, 2009). After taping the interviews, a transcription of thematically relevant passages was created. The transcription was less detailed as important elements were notated only to a certain extent, since a transcription of the whole recording is not standard (Bogner, Littig, and Menz, 2009).

The data were then coded using the Nvivo software, an effective tool for qualitative data analysis (Creswell & Creswell, 2018). In NVivo, extracted themes are presented as b nodes, with parent and child nodes for the main themes and subthemes, respectively. To identify the themes commonly discussed in the field, starting nodes were not predetermined. Nodes are also used as a tool to organize qualitative data in particular ways (Bazeley & Jackson, 2013), in this case, a comparison diagram is generated for child nodes users' data collection pros and users' data collection cons (Figure 2), showing cases where the four experts talked about the same issues (E1 – Expert 1, E2 – Expert 2, E3 – Expert 3, and E4 – Expert 4).

Figure 2: *A comparison diagram for child nodes 'Data collection pros' and 'Data collection cons'*



Source: Own elaboration, export from the analysis in the Nvivo software.

All four experts had a different style, so while three experts were talkative, one had to be guided and answered with semi-official statements for the most part. Because of the data structure, the text was sometimes coded line by line, sometimes sentence by sentence and sometimes even paragraph by paragraph. The main goal was to break down and understand the text and to attach and develop categories and put them into an order (Flick, 2009).

After the coding was done, thematically comparable passages from different interviews were tied together, and some coding was revised. In total, five themes were recognized (as presented in Figure 3): Users' data collection, Deciding on the redesign, A/B testing, User requirements, and Value of user satisfaction. It is in line with general recommendations for using coding to generate five to seven themes per research study (Creswell & Creswell, 2018). A larger area indicates more references, and the number of direct and aggregated coding references is specified as well.

Overall, the procedure consisted of two phases. The first one involved interviewing, after which the first concept of the framework was prepared and sent to the experts for a review. After considering their feedback, the revised version of the framework was proposed (Section 3.3).

Figure 3: *Hierarchy of nodes compared by the number of coding references*

3.2 Summary results of the interviews

3.2.1 Interview with Expert 1

The first interview was conducted with an (experienced) professional who is currently working as a quality assurance (QA) analyst and business analyst in a new IT company. Since the company just started operating, all company's projects are in the initial phase. It implies that there is very little or almost no data on existing users. In their case, the phase of collecting user requirements is very intensive, and sometimes more than half of the allocated project time goes into briefings. User requirements are documented, and most of them are reduced to functional requirements, i.e., the focus is mostly on processes than on the specific need that is to be supported by an application. Although there are some non-functional requirements, it usually gets down to a client's personal opinion, such as:

"This is ugly, it looks too complicated, the font is ugly, this image is too big, and the colors are ugly."

When asked about the automated collection of certain data and the prospects of its usage, the company plans to keep track of the technical logs and business metrics – but these are not yet defined. Also, the interviewee mentioned using Google Analytics as a tool and is aware that it provides more business metrics than other available data that can indicate some defect, which can possibly be used to improve UX. One of the further steps for the company will be the redesign of the current versions they are developing for their clients, so they certainly accept the idea of systematically incorporating data-based improvements into the next versions. Now, in the initial (demo) phase, technical metrics have already been set up, but there has not been an opportunity for using them yet. As an example of using such data, the expert mentioned:

"One minute for some action is maybe too much… well it does not have to mean that one minute is too much, but for an example, we will keep track of the average duration, so based on that we will make some improvements if necessary."

To improve user experience in the initial phase, they involved one expert for conversion who suggested some specific changes in the design. The proposal was to incorporate the padlock icons in the parts of the application where users make payments to emphasize the security of the web application. However, no one asked the end-users; they just assumed that this would help. Mentioning this specific example made the expert think about using the A/B testing method. The expert agreed on the importance of A/B testing, especially regarding some major features. However, they will not have two versions of the application running at the same time since it would be too expensive. The expert further got acquainted with the basic tools for passive monitoring, such as heat maps and session recording, and found them very useful, but still did not have the opportunity to incorporate these into their working process. At the end of the interview, the issue of GDPR was also discussed as one of the potential problems.

3.2.2 Interview with Expert 2

The second interview was conducted with another expert, who is currently working as the head of the programming department and has been working on that same position in the company for several years now.

When asked about collecting user requirements, the expert answered that concerning collecting information from the users, the company differs between functional and design requirements, but does not document them all – a similarity with the first expert's experience can be observed here. They rarely get non-functional requirements from their users, so they create a design conforming to the standard guidelines.

The expert stated that some data is automatically collected, but it mostly refers to server logs. They use those logs to check whether there has been some unexpected crash of the application. In case the crash is repeated, they reprogram and launch another version of the application. Even though without using any additional tools, the company could easily collect all usage data, such as the number of clicks, mouse tracks, time tracking, and more, but they still do not collect it. Another type of logs was mentioned, and those are the logs that show whether each user uses the application in general, but not the extent to which it is used.

They have never used A/B testing formally, even though they usually put an application to production, and they wait for the response from end-users. As an example, one of the versions was more complicated than the old one, which users reported themselves after they realized that it took them a lot more time to accomplish a certain task, as compared to previous versions. The company is aware that this may have a negative impact on users' satisfaction.

Although the expert agreed on the fact that data-oriented approach would have a better effect on UX, the interviewee focused rather on the disadvantages of this approach, such as the memory that this data would occupy. Additionally, related to this issue after connecting to a server, the expert added:

"Look at the amount of these logs and imagine how much data would be here if all click data were recorded? Do you know how many miles do you run with your mouse every day?"

However, as the interview continued, the importance of users' satisfaction was nevertheless to a certain extent acknowledged by the expert, as demonstrated in the following statement:

"It would be better if the data pointed out the problems that the user encountered without waiting for them to tell us."

By the end of the interview, the expert did agree that collecting this data for purposes of redesign would eventually shorten the time spent on conversations and interactions with end-users, especially after pointing out the fact that the main goal is still to keep users satisfied with the service and to improve their overall experience, particularly in this extremely competitive industry.

3.2.3 Interview with Expert 3

The third interview was conducted with a CEO and co-founder of a start-up, whose main activities are in digital marketing (a five-year experience) and in manufacturing natural cosmetics products. Due to her knowledge of digital technologies, together with a business partner they created a website with an integrated webshop. For developing their website, they did not collect any user requirements, since they decided to create it very spontaneously, and they wanted it done quickly. They followed their intuition combined with some standard guidelines, such as the ones related to the number of clicks, amount of text, picture resolution, and similar. They have put great emphasis on the visual appeal, since their primary goal is to establish a well-recognized brand. They believe that the visual part is crucial for attracting new customers and plays a huge part in their revenue, but the expert is also aware that without a functional website visitors would not buy their products.

As for external data collecting tools, they have been using Google Analytics from the beginning. They have recently participated in the research (which her former colleague was conducting), they have had Matomo tool installed (toll for recording users' sessions and creating Heatmaps). Even though the expert did not conduct A/B testing herself, their website was used for an experiment. For that purpose, an alternative website was created, and user data was collected and compared. The experiment results were unexpected.

"Research participants picked the alternative design as better (the results of objective data showed the same thing) …but it is because the participants were students who are not our target group …we target tourists, hipsters …the alternative website was based on a template, and I do not like it. It is very, very ugly."

Although objective data showed some potential for redesigning the current website, she did not take into consideration any changes in the visual aspect of the website. In that particular case, personal opinion and intuition prevailed over objective data. As for the functional changes, there are several things which they plan to change – they came as a result of A/B testing. Such as:

"The results of A/B testing were actually useful; it is just that we do not want to step away from our visual appeal. For example, testing showed that language plugin is better when positioned top right than on the bottom of the page."

In general, her attitudes towards collecting user data and data-driven (re)design were very positive, even though they plan to postpone this approach for a while:

"We plan to implement that approach, and I do believe it is the smartest thing to do, but the biggest problem for us currently is a lack of time. Besides, it is quite expensive. I hope that when we get to the stage where we can employ a person who will be in charge of that. We understand how important it is, but we currently have other priorities."

They do not formally manage user reviews, but since they still have a relatively small customer database, they have been contacting each customer to see how their shopping experiences were, and if they had any suggestions regarding improvements. They are not quite sure when they would take action, whether they would wait for some feedback to be reported repeatedly (such as their webshop being too complicated) or would they try to make improvments based on only one or two similar customer feedbacks.

As for the negative aspects of user data collection, besides time and cost (mentioned several times), the expert added the GDPR – or to be more precise, the unclear instructions for its implementation.

3.2.4 Interview with Expert 4

The last interview was conducted with another IT professional, who is currently working in a developer position with 10 years experience. Besides coding, he recently became a leader of a small team and also takes part in writing technical requirements. Collecting user specifications is very important, and they put a lot of effort into that phase, regardless of whether it is about upgrading the current system or developing a new one. Workshops are organized, where the most typical future users are gathered in the same place. During this phase, they use all data available, implicit and explicit:

"In the context of user feedback, the users' impression is always taken into consideration when developing a system… Also, we always do a detailed analysis of users' logs from our system – using tools customized for extracting key parts of the log, and moreover, we have a database where we can, with some classic queries, get usage data, for example, every month."

The company is very data-driven oriented; they have also developed a monitoring system:

"Monitoring observes live system data and enables real-time response to the needs of users."

The expert further explained the real example of how this monitoring system works. When observing the logs, they compare logs which are typical of normal usage, and if they notice any discrepancy, they do some modifications.

They have been applying data-driven approach from the beginning, but over the time, they have evolved since their systems became more robust. Now they put a lot more attention to user data. The expert confirms applying the data-driven approach in all kinds of situations (designing from the start, redesigning, and redesigning on request).

When asked about the methodology they follow, his answer was:

"It depends, everyone wants it to be a waterfall, but real-life situations require an agile approach."

When asked about the end-user satisfaction and experience, they do not have any form of tracking it, but they do get feedback from time to time. For example, they differ between two kinds of users, the ones that are agile and willing to spend their time to fix possible problems with the system, and the other ones that are inert.

They do not practice A/B testing due to the complexity of the systems they develop. Sometimes, while developing a new system or a feature, they provide end-users with several options – since they sometimes work with rapid development technologies. Also, some features/functionalities can be tested alone, but a high level of the analytical approach is hardly possible due to the complexity of their systems.

As the main advantage of a data-driven approach the expert sees in shortening time for both developers and end-users:

"Based on data, we noticed that our client had a business problem, which could be solved with minimum effort – so we made some changes. That was an example where we helped our client (even before he asked) and as a result got a satisfied user at a small cost. Mutual benefit..."

As for the disadvantages of this approach, the expert stated hardware resources and initial time needed to set it up, but, compared to the benefits, they are negligible. Even though experts in previous interviews mentioned the GDPR as a possible obstacle, the GDPR in his company does not present a problem, since they do not work on information systems for wide range usage. Data issues are covered within contracts with each client.

3.3 Discussion of the collected feedback

Even though the interviews followed a rigid methodological procedure, a small number of experts consulted (4) should be pointed out as one of the key limitations of this research. The results showed some observed differences (Table 3) between the experts, which were expected since they work in different settings. Expert 1 suggested that the interview could be repeated after a period of time when their application goes into production, since then the discussion about the type of data collected and similar would be more useful.

The interviews showed that there are also significant similarities between the experts' opinions. The most important is their agreement on the importance of incorporating the data-oriented approach into the whole IS development procedure. Also, they agreed that this approach, in their view, could improve the overall experience, without waiting for users' complaints. It could be especially useful in the redesigning projects, since without prior service usage there is no data available. Also, they agreed that there are still some unsolved issues and undefined gaps that have to be worked out to fully embrace the data-driven development paradigm.

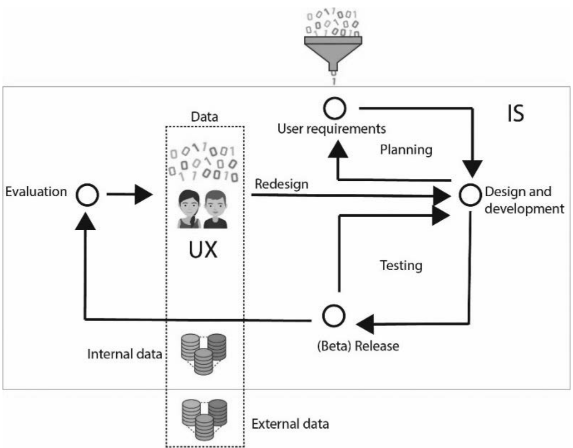
Table 3: *A comparison overview of the four conducted interviews*

Characteristics	Expert 1	Expert 2	Expert 3	Expert 4
Position	Quality assurance (QA) analyst and business analyst	Head of the programming department	CEO & co-founder of a start-up	Senior developer
Type of IS	Web applications	Web applications	Web applications	Web applications, desktop
Type of requirements collected	Functional, less non-functional	Functional, less non-functional	Functional	Functional and non-functional
Automatically collecting data	Not yet	Yes	Yes	Yes
Using data for decision making	No	No	Medium	Strong
Perception and attitude towards a data-oriented approach	Positive	Positive	Positive	Positive
Negative aspects	GDPR and other related issues	A lot of memory and time for such amount of data	GDPR, time-consuming, cost	
Focus on the satisfaction of end-users	High	Low-medium	Medium	High
User-oriented	Strong	Medium	Low	Strong
A/B testing use	Yes in the future, even if it is too expensive	Not formally, too expensive	Pilot	

As mentioned already, the first concept of the framework for user-oriented data-driven IS development was sent to the experts, and their feedback was received in the form of sketches and comments. After their comments, the changes and specifications were introduced, and the first version of the framework for user-oriented data-driven IS development is presented in Figure 4. To be more precise, several changes have been made in comparison with the first concept (Figure 1). First, there were several "basic" changes, such as adding "Planning", "Testing" and "(Beta) Release" for the more detailed presentation of the development process in the proposed framework. One more functional change has been made in terms of adding a new phase "Redesign", which now stands as a separate phase different from the previous "Development and redesign" phase to reflect its importance, and based on the comments from the experts, to better clarify the required types of input resources. Those input resources can be differentiated as "Internal" and "External" data, as well as "UX data". Another additional element of this framework is system boundaries. It is well known that the system by itself communicates with the environment, but this framework uses these boundaries to differ between internal data and external data. External data, as mentioned before, stands for objective passive data which is collected through some external web services, such as Google Analytics, HotJar, Matomo, and similar. Another change has been made – the relation between evaluation and user requirements has been removed, since after evaluation it is more fitting to define the next step as “Redesign” than start from the beginning.

The framework acknowledges the well-known phases of information systems development. The difference is that the proposed framework aims to emphasize UX and promotes using the data-driven approach systematically. Both in the theoretical and practical analysis with the experts, it became apparent that objective and passive data can be used effectively and frequently to improve UX by way of eliminating critical errors if these are detected or “just” in a way to improve the whole users’ journey while interacting with the system.

Figure 4: *The first version of the framework for user-oriented data-driven information systems development*



Source: Own elaboration.

In brief, the initial phase consists of collecting users' specifications. After that, the information system is being developed, which is then followed up by a testing phase. The testing phase is especially important for developing the first version of the system and, if done correctly, it can be supported by the collected usage data. To be able to use the data, metrics have to be defined and incorporated and devised in the development phase. A very important phase before any redesigning is using the collected data, as emphasized in the illustration.

The data that should be used for redesign purposes can come from different sources, external (tools used for passive tracking) or internal (server logs produced by users and by the system itself). Even though UX does not stand as some additional step in the proposed framework, by adopting the user-oriented and data-driven approach, UX should be improved as a result. The redesign process can also be repeated during the whole life cycle. As already mentioned, the agile approach aligns well together with these small and incremental changes and, as a result, a redesign can be done during several sprints. In general, the cycle of IS development never ends as long as the system is running.

4 CONCLUSION

A user-oriented data-driven information systems development process puts the needs of end-users in the center. To increase the use of digital services, users need to be satisfied. Two main issues were highlighted in the paper regarding users' needs – the inability to define who the end-user is and frequent changes in the requirements. The development and/or (re)design of an IS, regardless of the technology used, is never a simple process. In theory, the life-cycle should never end; whether it is about completely new versions or new updates of IS, a large number of applications are constantly updated even without the knowledge of end-users (Tomitsch, 2018).

Data collection through experimentation allows decision-making based solely on the needs of users, and such results provide the best possible experience for end-users (King, Churchill, and Tan, 2017). For this reason, the most commonly used method for mass experimentation and collection in the context above is A/B testing (Anderson, 2015; King, Churchill, and Tan, 2017) that was introduced in the first part of the paper and underlined during the interviews with the experts. It was presented to the experts based on relevant studies (Richre & Fluckiger, 2014; King, Churchill, and Tan, 2017; Tomitsch, 2018) primarily as a method that aims to compare two or more versions of an experience, a system, to detect differences, but at the same time enables the feedback loop and as such it is crucial for the user-oriented approach. Since it was well-received by the experts, further research on the proposed framework is planned to be more empirical in terms of developing several case studies to demonstrate the benefits of the data-driven development process and to devise and operationalize the specific elements of the framework, regardless of the data collection methods. Also, as many companies are using A/B tests routinely, more interviews with experts knowledgeable on the subject are planned, including the investigation on the pressing ethical issues (SciPol.org, 2019) regarding this specific method.

As of early 2019, serious efforts have been made towards passing the bill that would prohibit dividing individuals into groups for behavioral or psychological research without their consent, which could dramatically affect test practices. In that regard, the importance of collecting objective usage data would increase, and other methods would gain in popularity that would be analyzed in relation to the proposed framework. Apart from that, further research is planned to be done relating to the specific value of implicit data for the use as a part of the framework, as there is evidence that it can provide new insights beyond the usage patterns and user satisfaction (Stephens-Davidowitz, 2017). For that purpose, preparation of an experiment with one of the experts is ongoing and planned on top of the systems they are (re)designing. The aim is to compare the original version and the version extended with experimental functionalities while following the future detailed specifications of the framework to examine the effects and the feasibility of the approach in practice.

In the paper, a general notion of the importance of the objective, observable and available (big) data for the development of a user-oriented service or a system was presented followed by valuable inputs from the professionals in the field, all this to structure the first version of the future framework. Authors are fully aware of many issues in this area (some of which are published in Mijač, Jadrić, and Ćukušić, 2018) and would take these into account when developing and describing the specific elements of the framework in the following phases of the research.

ACKNOWLEDGEMENT

This work has been supported by the Croatian Science Foundation (grant CSF-IRP-2017-05-7625).

REFERENCES

Amokrane, N. et al. (2015). ‘Requirements authoring and verification for SMEs’ information systems engineering’, *IFAC-PapersOnLine.* Elsevier Ltd., 28(3), pp. 2238–2243. DOI: 10.1016/j.ifacol.2015.06.421.

Anderson, C. (2015). *Creating a Data-Driven Organization*. Sebastopol, Ca, SAD: O’Reilly.

Andrica, S. and Candea, G. (2011). ‘WaRR: A tool for high-fidelity web application record and replay’, *Proceedings of the International Conference on Dependable Systems and Networks*, pp. 403–410. DOI: 10.1109/DSN.2011.5958253.

Au, F. T. W., Baker, S., Warren, I., Dobbie, G. (2008) Automated usability testing framework. In Proceedings of the ninth conference on Australasian user interface - Volume 76 (AUIC '08), Vol. 76. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 55-64.

Bakaev, M., Mamysheva, T., Gaedke, M. (2016) Current trends in automating usability evaluation of websites: Can you manage what you can't measure?, 11th International Forum on Strategic Technology (IFOST), Novosibirsk, 2016, pp. 510-514. DOI: 10.1109/IFOST.2016.7884307

Balaji, S. and Sundararajan Murugaiyan, M. (2012) 'WATEERFALLVs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC', *International Journal of Information and Business Management, 2*(1), pp. 26–30. DOI: 10.1.1.695.9278.

Barry, B., & Wilfred, J. H. (2000). *Spiral Development: Experience, Principles and Refinements*. Pittsburgh: SEI Joint Program Office.

Bazeley, P. and Jackson, K. (2013). *Qualitative data analysis with NVIVO*. SAGE Publications.

Belanche, D., Casaló, L. V. and Guinalíu, M. (2012) 'Website usability, consumer satisfaction and the intention to use a website: The moderating effect of perceived risk', *Journal of Retailing and Consumer Services, 19*(1), pp. 124–132. DOI: 10.1016/j.jretconser.2011.11.001.

Bjarnason, E. et al. (2016) 'A multi-case study of agile requirements engineering and the use of test cases as requirements', *Information and Software Technology*. Elsevier B.V., 77, pp. 61–79. DOI: 10.1016/j.infsof.2016.03.008.

Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, (5), pp. 61-72.

Bogner, A., Littig, B. and Menz, W. (2009). *Interviewing Experts*. Palgrave Macmillan.

Brown, A., Fishenden, J. and Thompson, M. (2014). *Digitizing government*. New York, Palgrave Macmillan.

Brynjolfsson, E., Hitt, L. M. & Kim, H. H. (2011). Strength in Numbers: How Does Data-Driven Decisionmaking Affect Firm Performance? SSRN eLibrary. DOI: 10.2139/ssrn.1819486

Cohen, J. (2006). 'Social, Emotional, Ethical, and Academic Education: Creating a Climate for Learning, Participation in Democracy, and Well-Being', *Harvard Educational Review, 76*(2), pp. 201–237. DOI: 10.17763/haer.76.2.j44854x1524644vn.

Cormier, P. and Lewis, K. (2015). 'An affordance-based approach for generating user-specific design specifications', *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM, 29*(3), pp. 281–295. DOI: 10.1017/S089006041500027X.

Courtemanche, F. et al. (2017). 'Physiological heatmaps: a tool for visualizing users' emotional reactions', *Multimedia Tools and Applications*. Multimedia Tools and Applications, pp. 1–28. DOI: 10.1007/s11042-017-5091-1.

Creswell, J. W. and Creswell, J. D. (2018). *Research design*. 5th edn. London: SAGE Publications.

Espada, J., García Díaz, V., Gonzalez Crespo, R., Marín, C., Sanjuán, O., Pelayo García-Bustelo, B. et al. (2012). Method Based on Context-Information to Improve User Experience on Mobile Web-Based Applications. Advances in Artificial Intelligence - IBERAMIA 2012 - 13th Ibero-American Conference on AI, Cartagena de Indias, Colombia, November 13-16, 2012. Proceedings. 732-741. DOI: 10.1007/978-3-642-34654-5_74.

Fabijan, A., Dmitriev, P., Holmström Olsson, H. and Bosch, J. (2017) The evolution of continuous experimentation in software product development: from data to a data-driven organization at scale. In Proceedings of the 39th International Conference on Software Engineering (ICSE '17). IEEE Press, Piscataway, NJ, USA, 770-780. DOI: 10.1109/ICSE.2017.76.

Flick, U. (2009). *An introduction to qualitative research*. 4th edn. Sage Publications Ltd.

Frantz, T. L. (2018) 'Blockmap: an interactive visualization tool for big-data networks', *Computational and Mathematical Organization Theory*. Springer US, 24(2), pp. 149–168. DOI: 10.1007/s10588-017-9252-6.

Garcia, J. E., Paiva, A. C. R. (2016). An automated approach for requirements specification maintenance. *Advances in Intelligent Systems and Computing*, 444, 827–833. DOI: 10.1007/978-3-319-31232-3_78

Garcia, J. E., Paiva, A. C. R. (2016). Maintaining Requirements Using Web Usage Data, *Procedia Computer Science*. 100, pp. 626–633. DOI: 10.1016/j.procs.2016.09.204.

Garrido, A. et al. (2017). Data-driven usability refactoring: Tools and challenges, *SoftwareMining 2017 - Proceedings of the 2017 6th IEEE/ACM International Workshop on Software Mining, co-located with ASE 2017*, (October). DOI: 10.1109/SOFTWAREMINING.2017.8100854.

Geogy, M. and Dharani, A. (2016). A Scrutiny of the Software Requirement Engineering Process, *Procedia Technology*. 25, pp. 405–410. DOI: 10.1016/j.protcy.2016.08.125.

Gordillo, A. et al. (2014). The usefulness of usability and user experience evaluation methods on an e-Learning platform development from a developer's perspective: A case study, *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, pp. 1–8. DOI: 10.1109/FIE.2014.7044340.

Grigera, J. et al. (2017). Automatic detection of usability smells in web applications, *International Journal of Human Computer Studies, 97*(October 2017), pp. 129–148. DOI: 10.1016/j.ijhcs.2016.09.009.

Halvorsrud, R., Kvale, K. and Følstad, A. (2016). Improving Service Quality Through Customer, *Journal of Service Theory and Practice, 26*(6). DOI: 10.1108/JSTP-05-2015-0111.

Harrati, N. et al. (2015). Automating the evaluation of usability remotely for web applications via a model-based approach, *NTIC 2015 - 2015 1st International Conference on New Technologies of Information and Communication, Proceeding*. DOI: 10.1109/NTIC.2015.7368757.

Havice, J. (2017). *How to Create Customer Personas with Actual, Real Life Data*. Available at: http://bit.ly/2FrhRdx.

Heikkila, V. T. et al. (2015). A Mapping Study on Requirements Engineering in Agile Software Development, *Proceedings - 41st Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2015*, pp. 199–207. DOI: 10.1109/SEAA.2015.70.

Huang, Z. and Benyoucef, M. (2014). Usability and credibility of e-government websites, *Government Information Quarterly*. Elsevier Inc., 31(4), pp. 584–595. DOI: 10.1016/j.giq.2014.07.002.

Ilbahar, E. and Cebi, S. (2017). Classification of design parameters for E-commerce websites: A novel fuzzy Kano approach, *Telematics and Informatics*. Elsevier Ltd. DOI: 10.1016/j.tele.2017.09.004.

Inayat, I. et al. (2015). A systematic literature review on agile requirements engineering practices and challenges, *Computers in Human Behavior*. Elsevier Ltd, 51, pp. 915–929. DOI: 10.1016/j.chb.2014.10.046.

Inversini, A., Cantoni, L. and Bolchini, D. (2011). Design, User Experience, and Usability. Theory, Methods, Tools and Practice, 6770(May 2014). DOI: 10.1007/978-3-642-21708-1.

King, R., Churchill, E. F. and Tan, C. (2017). *Designing with Data.* O'Reilly.

Kujala, S., Kauppinen, M. and Rekola, S. (2001). Bridging the Gap between User Needs and User Requirements, *Advances in Human-Computer Interaction I (Proceedings of the Panhellenic Conference with International Participation in Human-Computer Interaction PC-HCI 2001)*, (February), pp. 45–50.

Lee, Y. Y. et al. (2016). Innovative Data-Driven Methods to Improve Digital User Experience., *Qualitative & Quantitative Methods in Libraries, 5*(2), pp. 461-471.

Lemon, K. N. and Verhoef, P. C. (2016). Understanding Customer Experience Throughout the Customer Journey, J*ournal of Marketing, 80*(6), pp. 69–96. DOI: 10.1509/jm.15.0420.

Liikkanen, L. A. (2016). *Tools for Data-Driven Design of Web Services*. Available at: https://www.linkedin.com/pulse/tools-data-driven-design-web-services-lassi-a-liikkanen.

Maalej, W. et al. (2015). Towards Data - Driven Requirements Engineering, 33, pp. 1–6. DOI: 10.1109/MS.2015.153.

Maguire, M. (2013). Using human factors standards to support user experience and agile design, *Lecture Notes in Computer Science*, 8009 LNCS (PART 1), pp. 185–194. DOI: 10.1007/978-3-642-39188-0-20.

Maguire, S. and Redman, T. (2006). The role of human resource management in information systems development, *Management Decision, 45*(2), pp. 252–264.

Marshall, B. et al. (2013). DOES SAMPLE SIZE MATTER IN QUALITATIVE RESEARCH?: A REVIEW OF QUALITATIVE INTERVIEWS IN IS RESEARCH, *Journal of Computer Information Systems*, pp. 11–22.

Medeiros, J.D.R.V., Alves, D.C.P., Vasconcelos, A., Silva, C., Wanderley, E. (2015). Requirements engineering in agile projects: A systematic mapping based in evidences of industry. CIBSE 2015 - XVIII Ibero-American Conference on Software Engineering, pp. 460-473.

Mijač, T., Jadrić, M. and Ćukušić, M. (2018). Evaluating the Potential of a Data-Driven Approach in Digital Service (Re) Design, Proceedings of the Central European Conference on Information and Intelligent Systems, pp. 187–194.

Oertel, K. and Hein, O. (2003). Identification of Web usability problems and interaction patterns with the RealEYES-iAnalyzer, *Interactive Systems Design, Specification, and Verification 10th International Workshop, DSV IS 2003 Revised Papers Lecture Notes in Comput Sci Vol 2844*, 2844, pp. 77–91.

Olsson, H. H. and Bosch, J. (2014). The HYPEX model: From opinions to data-driven software development, *Continuous software engineering*, 9783319112(August), pp. 155–164. DOI: 10.1007/978-3-319-11283-1-13.

Pietkiewicz, I. and Smith, J. A. (2012). A practical guide to using Interpretative Phenomenological Analysis in qualitative research psychology 1, *Czasopismo Psychologiczne*, pp. 361–369. DOI: 10.14691/CPPJ.20.1.7.

Reddick, C. G. and Roy, J. (2013). Business perceptions and satisfaction with e-government: Findings from a Canadian survey, *Government Information Quarterly.* Elsevier Inc., 30(1), pp. 1–9. DOI: 10.1016/j.giq.2012.06.009.

Richre, M. and Fluckiger, M. (2014). *User-Centred Engineering*. Schlieren, Switzerland: Springer.

Robey, D. and Farrow, D. (1982). User Involvement in Information System Development: A Conflict Model and Empirical Test, *MANAGEMENT SCIENCE, 28*(1), pp. 73–85.

Rodden, K., Hutchinson, H. and Fu, X. (2010). Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, *SIGCHI Conference on Human Factors in Computing Systems*, pp. 2395–2398. DOI: 10.1145/1753326.1753687.

Rodriguez, M. G. (2002). Automatic data-gathering agents for remote navigability testing, *IEEE Software, 19*(6), pp. 78–85. DOI: 10.1109/MS.2002.1049396.

Saadawi, G. M. et al. (2005). A method for automated detection of usability problems from client user interface events. *AMIA Symposium*, (3), pp. 654–658.

SciPol.org (2019). Summary for Deceptive Experiences To Online Users Reduction Act, Duke Initiative for Science & Society, available at https://scipol.org/track/s-1084-deceptive-experiences-online-users-reduction-act

Sengers, P. et al. (2008). The disenchantment of affect, *Personal and Ubiquitous Computing, 12(*5), pp. 347–358. DOI: 10.1007/s00779-007-0161-4.

Smith, J. A. and Osborn, M. (2007). Interpretative Phenomenological Analysis, *Qualitative Psychology*, pp. 53–80. DOI: 10.1002/9781119975144.ch9.

Sohaib, O. and Khan, K. (2010). Integrating usability engineering and agile software development: A literature review, *2010 International Conference On Computer Design and Applications*, 2(Iccda), pp. V2-32-V2-38. DOI: 10.1109/ICCDA.2010.5540916.

Spiess, J. et al. (2014). Using Big Data to Improve Customer Experience and Business Performance, *Bell Labs Technical Journal, 18*(4), pp. 3–17.

Stephens-Davidowitz, S. (2017). *Everybody lies*. Dey Street Books.

Svensson, R. B., Feldt, R. and Torkar, R. (2019). The Unfulfilled Potential of Data-Driven Decision Making in Agile Software Development, in Kruchten, P., Fraser, S., and Coallier, F. (eds) *Agile Processes in Software Engineering and Extreme Programming*. Cham: Springer International Publishing, pp. 69–85.

Tomitsch, M. (2018). *Making cities smarter*. Berlin: Jovis.

Vanwersch, R. J. B. et al. (2015). A Critical Evaluation and Framework of Business Process Improvement Methods, *Business & Information Systems Engineering. Springer Fachmedien Wiesbaden, 58*(1), pp. 1–11.

Wiesner, S., Nilsson, S. and Thoben, K. Di. (2017). Integrating Requirements Engineering for Different Domains in System Development - Lessons Learnt from Industrial SME Cases, *Procedia CIRP*. 64, pp. 351–356. DOI: 10.1016/j.procir.2017.03.013.

Zarour, M., & Alharbi, M. (2017). User experience framework that combines aspects, dimensions, and measurement methods. *Cogent Engineering, 4*(1), 1421006. DOI: 10.1080/23311916.2017.1421006.