Converging Data with Design Within Agile and Continuous Delivery Environments

Jay Brewer, Ger Joyce(✉), and Saurabh Dutta

Rapid7, 100 Summer Street, Boston, MA 02110, USA
{jbrewer,gjoyce,sdutta}@rapid7.com

Abstract. Traditional user research methods, while vital to understanding software application users, can be slow to implement and learn from. Even after results are analyzed, behavioral intention, not actual user behavior, tends to be better understood. To complement this approach, big data and behavioral analytics can be used to quickly learn more about actual user behavior, thus converging data with design. This is important within fast-paced agile and continuous delivery environments. At Rapid7, an IT and analytics information security organization, the convergence of data with design has allowed for insights that have informed design decisions, which have met users' mental models and actual needs. To that end, the software design industry is starting to diverge from the form-follows-function school of design, whereby the conventional approach of the user informing the designer has been reversed.

Keywords: Data · Design · User research · UX · Human-Computer interaction

1 Introduction

Traditional user research methods, such as surveys, focus groups, contextual inquiry, and interviews (Fig. 1), can be time-consuming to set up, run, and analyze [1]. In addition, traditional methods tend to elicit behavioral intention, which may differ to actual user behavior once a product or feature is released. This approach was well-suited to slower paced waterfall software development processes. However, as fast-paced agile and continuous delivery software development processes become ubiquitous [2], the design of software applications is rarely static—applications across all channels, including the cloud, mobile, desktop, virtual reality, and wearables, tend to change, even minimally, with each iteration. Thus, Human-Computer Interaction (HCI) researchers are finding that there is little time for research before software is released. While the quantitative methods and qualitative methods they have relied upon for years are not going away anytime soon, researchers need a method that will allow them to inform design decisions quickly.

Enter big data and behavioral analytics—both have changed the face of Human-Computer Interaction (HCI) research, and will become the mainstay of researchers in the years to come. This paper looks deeper into this phenomenon, where data converges with design within agile and continuous delivery environments. By gleaning insights from big data and behavioral analytics, HCI researchers can answer many of their questions, even before speaking with a single user of an application. This is a

A. Marcus and W. Wang (Eds.): DUXU 2017, Part I, LNCS 10288, pp. 533–542, 2017.
DOI: 10.1007/978-3-319-58634-2_39

Fig. 1. An author of this work conducting traditional HCI research

relatively new approach, which can be beneficial to many industries, not least information security. With many of the world's top organizations fully reliant on information security software, it is simply not good enough for information security software providers to release updates months or years apart. One of the leading information security software providers, Rapid7, an organization that provides software to transform data into insight for IT and security professionals, has ensured that their HCI designers have full access to big data and behavioral analytics, thus allowing data to inform design decisions.

Within this paper, the authors focus on an empirical example of the convergence of data and design at Rapid7. During the case study, the design implications from behavioral analytical learnings, which may not otherwise have occurred, are discussed. The information security application that we will focus on is InsightIDR, an incident detection and response tool. This application detects and alerts on information security related incidents and suspicious behavior, such as that caused by stolen employee user names and passwords.

2 Background

2.1 Waterfall Software Development Methodology

The waterfall software development methodology was the primary approach to developing software in the 1980's and 1990's. The methodology is linear in nature (Fig. 2), whereby each phase could not start until the previous phase was fully completed. Requirements were fully laid out months or years in advance of the final

delivery of the product. These requirements could take quite some time to define, given that there could be many stakeholders, and discussions around requirements were often lengthy. Once requirements were finally defined, only then did the design phase start. Given the relatively slow pace of waterfall software development methodologies, HCI researchers and designers usually had time to conduct a significant amount of research, as well as explore different design approaches. After the research and design deliverables were handed over to engineers, development started. Later in the process, Quality Assurance teams tested the software, and bugs were fixed. Following a review by the various stakeholders, the application was released to customers.

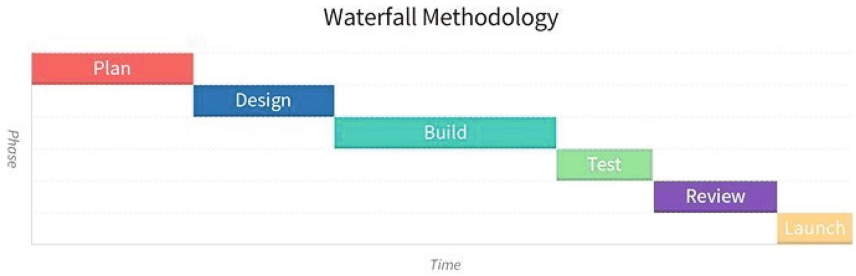

Fig. 2. Waterfall software development methodology

Even readers unfamiliar with the waterfall process will immediately see the potential issues. One such example was that as estimating project time and cost years in advance is quite difficult, most waterfall-based software development projects suffered high cost overruns, and were delivered much later than originally planned [3]. Additionally, during the months and years that the software was being developed, customer needs often changed. However, as a significant effort had been made in resources, time, and cost to create the software, major changes were rarely made. To that end, customers frequently did not receive the product they expected, resulting in dissatisfaction.

2.2 Agile Software Development Methodology

While waterfall was the primary methodology used for decades, changes needed to be made as software became more customer-focused. Consequently, the agile manifesto was created by a group of developers in Salt Lake City, Utah [4]. The new methodology promised a much shorter design and development loop (Fig. 3), with software being released usually every 1–4 weeks. This, in turn, allows for a much faster release cycle.

Yet, agile methodologies are not without problems. HCI researchers often have difficulty in conducting meaningful, rigorous research within fast-paced agile environments [5, 6]. In fact, common HCI-related activities, such as usability evaluations, are often eliminated within fast-paced agile environments, [7, 8]. It is possible that this was foreseen as the agile manifesto was being developed, however it may not have

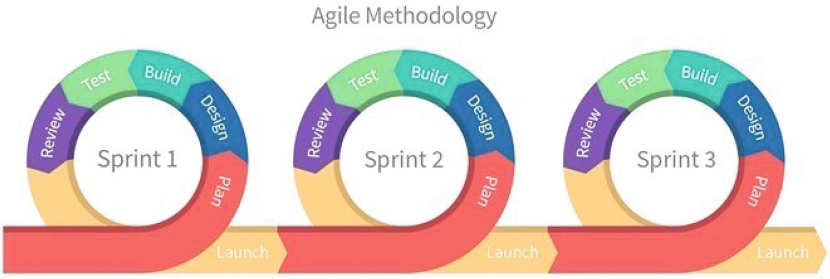

Fig. 3. Agile software development methodology

been considered an issue. The original view of those that created the agile manifesto was to release early and often to customers, then make changes based on their feedback, which meant, in theory at least, HCI research was no longer required. However, this could lead to a poor user experience as releases may not meet customer needs (even if initially during the first version), and where usability is sub-standard as the software is developed by engineers, not designed by HCI experts. The result could be customers that perceive a product to be unusable and/or not useful. Even though subsequent releases might be better due to customer feedback, negative online reviews could result in poor future sales [9]. HCI researchers, therefore, have a vital role to play within agile environments. However, HCI researchers must adapt their approach, even with traditional research, in order to offer actionable insights as quickly as possible. In short, HCI researchers need to plan, gather, and analyze research much faster within agile environments [10].

2.3 Continuous Delivery Software Development Methodology

Agile methodologies result in releases every several weeks. Therefore, while agile software development methodologies are faster than waterfall, many organizations find that even this release cadence this too slow to deploy software within an ever-increasingly competitive market. Consequently, several pioneers have automated many aspects of the software development life cycle, such as testing and building. This allows for small incremental releases, whereby bug fixes, new features, backend updates can be released quickly, often on a daily basis. This software development methodology has become known as continuous delivery (Fig. 4) [11]. While the practice is not yet widespread, the promise of faster time-to-market, fewer software bugs, and more secure applications [12] may lead to a growth in popularity. This is despite the significant technical challenges involved in setting an organization up for continuous delivery [13].

The organizations that have implemented the methodology report good results. Several teams at Facebook, for instance, have implemented continuous delivery software development methodologies, which have resulted in thousands of incremental deployments every day [2]. Even with this high number of deployments, the quality of the software actually rose.

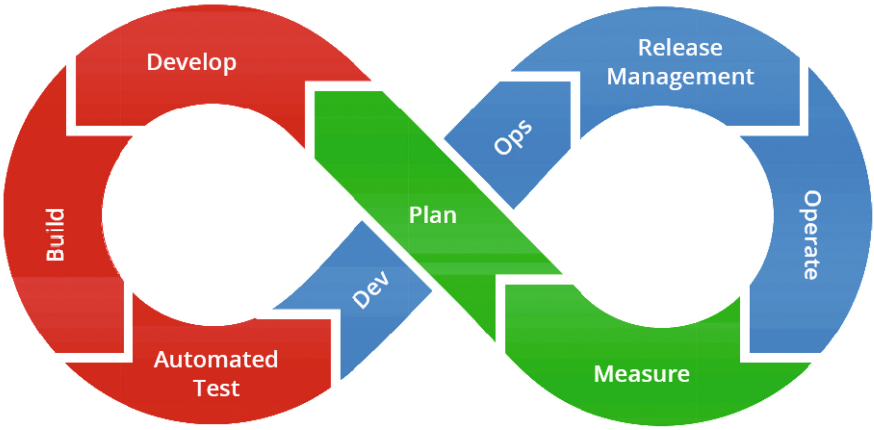

Fig. 4. Continuous delivery software development methodology

It is no surprise that HCI researchers working within continuous delivery environments have found that the time for up-front research, as well as discovering changes in user behavior, has diminished due to the increase in frequency of software releases. Researchers have attempted to tackle this challenge in several ways. For instance, Beasley [14] investigated the use of Google Analytics, yet, concluded that such data was designed for marketing teams, not HCI experts. From a mobile perspective, McGregor et al. [15], recorded participants' screens, only to discover that analyzing the resulting videos was prohibitively slow. As more and more organizations migrate to fast-paced methodologies, the convergence of data with design will become even more important.

3 Case Study

To focus on an empirical illustration of the convergence of data with design, consider InsightIDR, Rapid7's incident detection and response cloud-based application. The product was designed to assist incident responders, who are generally notified with an email alert, in surfacing data surrounding a potential information security breach. Should the probability lean toward an actual information security breach, an incident responder could use the application to add context, such as an attacker's path, by using related enriched data from within and outside Rapid7 applications. The full scope of the issue could be defined within an investigation timeline within the application. The end-result being that the attacker's path can be quickly shut down, and sealed against future attack.

As is common with software application design, traditional user research methods were employed during the initial research phase. This took the form of focus groups, surveys, and interviews with information security professionals. As the software was developed and released on a fast-paced agile schedule, the research team continued to rely on traditional research methods. However, the pace of research and development

were not aligned. As a focus group, for instance, was being organized, the research team often found that a feature had already been released by the development team. While this did not mean that the focus group was cancelled, it did mean that faster-paced user research methods needed to be considered. To that end, the HCI team at Rapid7 started to rely heavily on big data and behavioral analytics. With limited resources on the team, this became vital.

While, much of the data gathered using both traditional and contemporary research methods are analogous, the differences between behavioral intention and actual user behavior were discernible. For instance, during the focus groups and interviews conducted prior to the initial release of an Investigations feature, most participants stated that they would use the proposed new feature often. The Investigations feature had the benefit of collating disparate, eclectic data to better form a series of actionable insights. However, several months following release of the feature, analysis of behavior analytics uncovered a startling insight—the Investigations feature had been little used by any customers since its release. When a feature is not being utilized as much as it might, this may not be critical for many industries. However, underutilized information security applications could decrease the security posture of an organization. This insight resulted in an informed re-design to ensure that the feature was even more visible, useful, and useable. The re-design led to a three-fold increase in the use of the Investigations feature (Fig. 5).

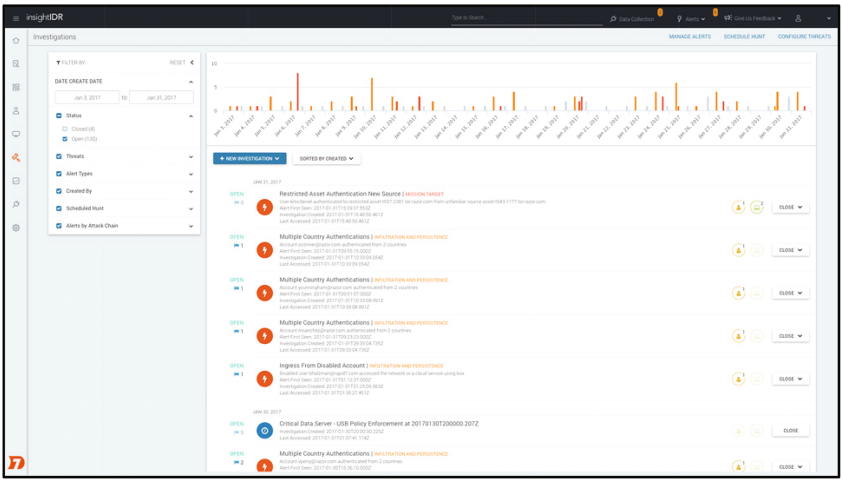

Fig. 5. InsightIDR investigations

When behavioral analytic insights show that a feature is underutilized, there are other strategies that can be employed, should an increase in usage be deemed critical. One such strategy is to tactically inform a user at the appropriate time that it might be in their best interest to consider the feature. In the case of InsightIDR, this approach led to more information security professionals closing Investigations (Fig. 6). While this is never a substitute for well-designed software, the approach can help HCI teams to

guide the users of their applications to success. The messages employed not only need to be tactful, but they should also not be intrusive. Tactful nudges should only be used to guide a user to complete a task, or allow the user to consider a more effective approach that they might not otherwise have considered. Such guidance, therefore, should not clutter or adversely impact the user experience. Another such campaign within InsightIDR resulted in a four-fold increase in adoption of dashboards. These nudges were initially a manual effort, and have since been automated. This approach is scalable and cost-effective in that it is conducted in a touchless manner [16].

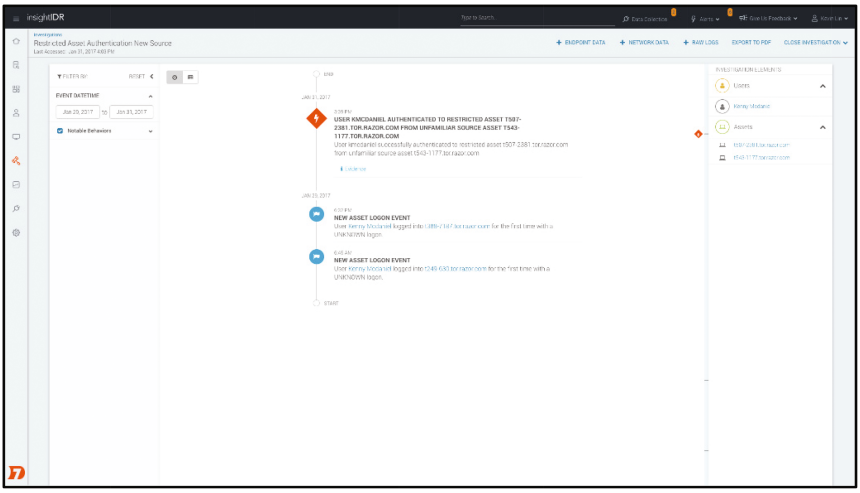

Fig. 6. InsightIDR: Closing an investigation

Other insights gathered using behavioral analytics included learnings around the user base. In some cases, software can be designed in different ways to suit different types of users. Alternatively, there could be one application that needs to be all things to many types of users. Each approach can be difficult to design for, even if each type of user is mapped to a specific persona [17]. While, the research team at Rapid7 already had defined a strong ecosystem of personas using Lean UX principles, it was only by using big data and behavioral analytics that the team became fully aware of which types of personas used which features. These insights were aggregated over time in the continuum of product usage. Additionally, knowing which features were used by which personas allowed the design of specific features for those personas that might not otherwise have been designed. Further, for future releases, the team's designs are tested by a segmented audience represented by a specific persona, generally during a canary release [18]. The insights gathered help to demonstrate if the design works or does not work for that audience. These insights are combined with the theory of planned behavior [19], allowing the design team to avail of predictive modeling. Based on inductive reasoning from predictive models, it can be determined if a feature will be used or not. Thus, the design of InsightIDR may change during subsequent iterations or features and functionality may simply be removed to save cost.

Yet, not only are insights gained in regard to user types. Access to applications in a seamless way are often sought by users across different types of devices and channels. This could include mobile, TV, wearables, and virtual reality. Using benchmarking and signals, changes in behavior can be detected regarding the switch from web to mobile and vice versa. With InsightIDR, this has led to the ability for users to receive actionable alerts on their mobile devices, as they are not always at their workstations. In a world where information security breaches are commonplace nowadays, minutes often matter to information security professionals. Actionable mobile alerts can, therefore, allow for the quick containment of an attack.

Finally, while there have been numerous advantages to complementing, comparing, and contrasting traditional user research with big data and behavioral analytics for the HCI team at Rapid7, the benefits do not end there. The HCI team could more easily share proven insights, with associated evidence, across the organization. This has led to previously siloed departments communicating regularly to ensure customer success, by enabling customers to fill the security gaps that they might not have realized existed. As sharing and reading qualitative data can be time-consuming, processes at Rapid7 needed to change to enable faster response times. Consequently, the HCI team collaborated with other teams to define the Product Engagement Index (PEI). The PEI assigns a single quantitative score, supported by qualitative insights, to aggregate user behavior. Thus, teams across Rapid7 can see how "healthy" a customer is based on their usage. A low PEI could signify a customer that underutilizes InsightIDR and other Rapid7 products due to a lack of security maturity and insufficient information security team skills. The Customer Success team could then suggest that the customer avail of Rapid7's Managed Services to plug the security gaps that existed within the customer's security posture. A medium PEI score could alert the Customer Success team to a maturing security customer that simply did not know about features that would increase their security maturity and posture. Tactful use of touchless guides, followed by a timely phone call from the Customer Success team, has proven to truly assist customers in becoming far more secure. This approach generally results in a higher PEI score, and in some cases over 200% increase in the usage of certain features. Additionally, PEI can also help to inform design decisions, as it is used to apprise aggregate focus for the overall design of InsightIDR and other Rapid7 tools.

4 Conclusion

Traditional research methods, such as surveys, focus groups, contextual inquiry, and interviews, tend to be slow to implement and analyze. While this was appropriate for slower-paced waterfall software development methodologies, with the advent of faster-paced agile and continuous delivery software development methodologies, it is important to consider more suitable options. This is due to the difficulty that HCI researchers have in finding the time for research before software is released.

One of the potential options, which can allow HCI researchers to learn more about which types of users are interacting with an application, as well as which tasks they are performing, is the use big data and behavioral analytics. From a philosophical perspective, the use of big data and behavioral analytics transforms HCI researchers'

epistemological perspective from more traditional paradigms, such as post-positivism and interpretivism, to a more pragmatic approach. Pragmatism [20] allows researchers to use whatever methods they feel will answer their research questions [21, 22]. This is an important distinction, as without a philosophical consideration, many researchers simply follow the most frequently used paradigm of their field [23]. This philosophical approach allows for the use of traditional research methods, in tandem with behavioral analytics, not a case of one or the other.

The use of big data and behavioral analytics has many benefits for HCI teams, allowing for the convergence of data and design. At Rapid7, the HCI team has relied on behavioral analytics primarily for three intrinsic benefits:

1. **User profiling and expertise assessment**
   Quickly and accurately learn about actual user behavior, so that custom experiences can be offered based on expertise and user type
2. **Predictive analytics**
   Better understand future user intentions, thus meeting precise user needs. This differs to traditional user research methods, which are often reactive
3. **Benchmarking and signals**
   Analyze trends across multiple channels, including web, mobile, wearable, TV, virtual reality, and Internet of Things (IoT) to establish norms and detect deviation across devices and channels.

Insights allow for informed design decisions whereby an application or parts of the application, can be quickly re-designed to meet users' mental models and actual needs, which is vital in a world of fast-paced agile and continuous delivery environments. Alternative options, however, should not replace traditional methods—instead, these options should be used to triangulate, complement, compare, and contrast holistic insights gathered. This is due to the efficiency of big data and behavioral analytics in answering "what"-type questions, but not the "why"-type questions. Consequently, it is often necessary to follow up on interesting behavioral analytics data with user interviews. These help HCI researchers to better understand why a user took a specific action. Even with the use of traditional research methods in answering some of the questions that behavioral analytics cannot, at Rapid7, we are aware that there is now a distinct divergence from the form-follows-function school of design, whereby the conventional approach of the user informing the designer has been reversed.

References

1. Salah, D., Paige, R., Cairns, P.: A practitioner perspective on integrating agile and user centred design. In: Proceedings of the 28th International BCS Human Computer Interaction Conference (HCI 2014), pp. 100–109 (2014)
2. Rossi, C., Shibley, E., Su, S., Beck, K., Savor, T., Stumm, M.: Continuous deployment of mobile software at facebook (showcase). In: Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE 2016), pp. 12–23 (2016)
3. Bloch, M., Blumberg, S., Laartz, J.: Delivering large-scale IT projects on time, on budget, and on value. Financial Times (21 August 2012). https://www.ft.com/content/d34acf86-eba8-11e1-9356-00144feab49a. Accessed

4. Agile Manifesto: (2001). Retrieved from http://agilemanifesto.org/
5. Haikara, J.: Usability in agile software development: extending the interaction design process with personas approach. In: Concas, G., Damiani, E., Scotto, M., Succi, G. (eds.) XP 2007. LNCS, vol. 4536, pp. 153–156. Springer, Heidelberg (2007). doi:10.1007/978-3-540-73101-6_22
6. Humayoun, S.R., Dubinsky, Y., Catarci, T.: User evaluation support through development environment for agile software teams. In: Smart Organizations and Smart Artifacts, pp. 183–191 (2014)
7. Wale-kolade, A.Y., Nielsen, P.A.: Integrating usability practices into agile development: a case study. In: 23rd International Conference on Information Systems Development (ISD 2014), pp. 337–347 (2014)
8. Raison, C., Schmidt, S.: Keeping user centred design (UCD) alive and well in your organisation: taking an agile approach. In: Marcus, A. (ed.) DUXU 2013. LNCS, vol. 8012, pp. 573–582. Springer, Heidelberg (2013). doi:10.1007/978-3-642-39229-0_61
9. Forman, C., Ghose, A., Wiesenfeld, B.: Examining the relationship between reviews and sales: the role of reviewer identity disclosure in electronic markets. Inf. Syst. Res. **19**(3), 291–313 (2008)
10. Salvador, C., Nakasone, A., Pow-Sang, J.A.: A systematic review of usability techniques in agile methodologies. In: Proceedings of the 7th Euro American Conference on Telematics and Information Systems (EATIS 2014), pp. 1–6 (2014)
11. Humble, J., Farley, D.: Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation. Pearson Education, Upper Saddle River (2011)
12. Chen, L.: Continuous delivery: overcoming adoption obstacles. In: Proceedings of the International Workshop on Continuous Software Evolution and Delivery (CSED 2016), p. 84 (2016)
13. Claps, G.G., BerntssonSvensson, R., Aurum, A.: On the journey to continuous deployment: technical and social challenges along the way. Inf. Softw. Technol. **57**(1), 21–31 (2015)
14. Beasley, M.: Practical Web Analytics for User Experience: How Analytics Can Help You Understand Your Users. Morgan Kaufmann, Burlington (2013)
15. McGregor, M., Brown, B., McMillan, D., Chi, W.: 100 days of iPhone use: mobile recording in the wild. In: Chi 2014, pp. 2335–2340 (2014)
16. Brewer, J., Dutta, S., Kriskovic, M.: From dark patterns to angel patterns: creating trust-worthy user experience. User Experience Mag **16**(5) (2017). Retrieved from http://uxpamagazine.org/from-dark-patterns-to-angel-patterns/
17. Bhattarai, R., Joyce, G., Dutta, S.: Information Security Application Design: Understanding Your Users. In: Tryfonas, T. (ed.) HAS 2016. LNCS, vol. 9750, pp. 103–113. Springer, Cham (2016). doi:10.1007/978-3-319-39381-0_10
18. Neely, S., Stolt, S.: Continuous delivery? Easy! just change everything (well, maybe it is not that easy). In: Proceedings AGILE 2013, pp. 121–128 (2013)
19. Ajzen, I.: The theory of planned behavior. Organ. Behav. Hum. Decis. Process. **50**, 179–211 (1991)
20. Talisse, R., Aikin, S.: Pragmatism: A Guide for the Perplexed. Bloomsbury Academic, London (2008)
21. Denscombe, M.: Good Research Guide: For Small-Scale Social Research Projects, 4th edn. McGraw-Hill Education, Berkshire, GBR (2010)
22. Frost, N.: Qualitative Research Methods in Psychology. McGraw-Hill, Berkshire, GBR (2011)
23. Hesse-Biber, S.N.: Mixed Methods Research: Merging Theory with Practice. Guilford Press, New York (2010)