Improvement of Novice Software Developers' Understanding about Usability: The Role of Empathy Toward Users as a Case of Emotional Contagion

Fulvio Lizano and Jan Stage

Aalborg University, Department of Computer Science, Aalborg, Denmark
{fulvio,jans}@cs.aau.dk

Abstract. There are several obstacles when it comes to integrating Human-Computer Interaction (HCI) activities into software development projects. In particular, a lack of understanding on the part of novice software developers regarding usability is one of the most cited problems related to this integration. Observation of usability evaluation by these developers has been cited in the literature as an alternative to improve their understanding about usability due to the fact that, among other things, this improves the level of empathy with users. In this paper we present the results of a quasi-experiment which explores the origin of this improvement. Our study suggests that the empathy of novice developers towards users could be originated by Emotional Contagion (EC) of these developers. This EC occurs unconsciously in activities where these developers can observe users working with the software. The present research is an initial approximation as to the relation which EC and empathy have in order to improve the novice software developers' understanding of usability.

Keywords: Software development, usability, understanding of usability, empathy towards users, emotional contagion.

1 Introduction

The lack of understanding on the part of novice software developer regarding usability, is one of the most cited problems about to integration of HCI activities (specially usability evaluations), into software development projects. [2, 3], [15], [18]. This problem suggests a low priority of software developers on the user. Developers' motivators confirm their focus on personal matters [9], [14].

According some studies, observation of usability evaluations by developers improves their understanding of usability and also their empathy with users [10], [21]. Other researchers confirm this increasing of empathy in contexts with close interaction with users [6, 7], [12, 13]. Causes of such phenomenon in developers have not been studied yet.

The empathy [5], [20] has its origin in an Emotional Contagion (EC) process [8], [17]. This process occurs between two actors: the observer and the observed. In the pro-cess, the observer unconsciously acquires the emotions of the observed after

M. Kurosu (Ed.): Human-Computer Interaction, Part I, HCII 2014, LNCS 8510, pp. 207–218, 2014.

seeing and interacting with him for some time [4], [20]. The observer assumes a submissive role in her/his interaction with the observed who, in turn, assumes a dominant role. The particular circumstances or personalities of each are decisive in establishing who assumes a particular role [17]. The EC-Process could be fundamental to explain why developers experiment an increasing in the empathy with users and also in the under-standing of usability, during their observation of usability evaluations.

Considering this, we conducted a quasi-experiment [23] which aimed to explore the improvement of the understanding of usability and also the empathy with users by novice software developers, into a usability evaluation context. Our study attempts to fill the gap in the literature by explaining this situation since a perspective of an EC-Process.

In the first section of this paper we present the introduction and a brief literature review. Next, the method is presented in the section 2. Following this, we present the results of our study. After the results have been summarized, the paper presents the discussion section before concluding with suggestions for future work.

2 Method

We conducted a quasi-experiment [19], [23] where nine developers (SE/CS students), grouped in two teams, conducted a usability evaluation with users [16]. Usability evaluations were used to set an interactive environment with users; our focus was on the improvement process of the understanding of usability, more than in the results of the tests.

We collected data related to the students' understanding of usability two weeks before the test (1DC) and immediately after (2DC) the test. Additionally, we held interviews with students. The aim of these interviews was to allow the authors to elaborate on or clarify some findings of the study.

In every DC we used two forms. The first form (F1) was used in order to allow the students to express their opinions related to the main strengths and weaknesses presented in their software. The second form (F2) was used to measure the relative importance given by the students to certain software/usability concepts. In this form, we used 5 pairs of concepts or sentences which could illustrate normal activities for SE or HCI practitioners.

The concepts related to SE were:

- Modelling software requirements.
- Understanding how a system is designed.
- Realizing how the Unified Modelling Language (UML) could be applied to a software project.
- Knowing about software modelling patterns.
- Understanding the main concepts of Object-Oriented modelling.

The concepts related to HCI were:

- Designing an interface both physically and conceptually correct.
- Understanding how a user interface could be designed.
- Realizing how the Gestalt Laws could be applied to a software project
- Knowing about visual design principles.
- Understanding the main concepts of Human-Computer Interaction.

The analysis of the data collected was focused on the identifying the improvement in the understanding of usability by analyzing differences (between 1DC and 2DC) in F1. In addition, we identified the understanding pattern of usability based on [1]. Results were triangulated with F2 and the interviews. As part of the analysis, we identified the origin of such improvement and the implications for the empathy toward users.

3 Results

In this section we present the results of the study. We felt that in order to better understand the mechanism(s) of generation of empathy towards users, we first needed to establish in a general and detailed way, beyond doubt, a real improvement in the understanding of usability. Following this, we could identify and understand better the patterns which characterized this improvement. This explains why we first focused on describing the variations in the understanding of usability after applying the corrective action (conduction of usability evaluation by the students); these results are presented in Sections 3.1, 3.2 and 3.3. Next, in Section 3.4 we will present the patterns which characterized the improvement of the novice software developers' understanding of usability. In this part of the results, we also included some of the students' personal opinions given during the interviews, in order to complete the picture.

3.1 Overall Understanding of Usability

We were interested in gauging the perceptions of students before and after their participation in the usability evaluation Table 1 presents the general results obtained when we enquired about the strengths (S) and the weaknesses (W) of their software (form F1).

Table 1. Strengths (S) and weaknesses (W) related to usability before and after conducting usability evaluation

Facts	F1				Variance	
	1 DC		2 DC			
	S	W	S	W	S	W
Total opinions (software + usability)	40	37	37	48	-3	+11
Opinions related to usability	16	12	11	37	-5	+25
Percentage	40%	32%	30%	77%	-10%	+45%

During 1DC the students provided 40 strengths and 37 weaknesses. 16 strengths were related to usability issues (40%). In addition, they provided 12 weaknesses (32%). In the 2DC the students provided 37 strengths and 48 weaknesses. In this case, 11 strengths were related to usability issues (30%) and 37 weaknesses were related to usability (30%). After the conduction of the usability evaluation the strengths related to usability decreased 10% whilst the weaknesses increased 45%.

The results of the relative importance given by the participants to software or usability matters (form F2) confirmed their perception about strengths and weaknesses. After the usability evaluation, the students' opinions changed in order to consider the usability as more important. It seems that usability becomes more relevant for students after they conduct the usability evaluation. These results are presented in Table 2.

Table 2. Strengths (S) and weaknesses (W) related to usability before and after conducting usability evaluation

Facts	F2			
	1 DC		2 DC	
	Related to software	Related to usability	Related to software	Related to usability
Favorable opinions	37	8	30	15
Percentage	82%	18%	67%	33%

3.2 Detailed Understanding of Usability

In Table 3, we present the strengths and weaknesses provided by students in 1DC and 2DC, which are related to usability. This table also includes the variation presented in these aspects after the usability evaluation.

Table 3. Strengths (S) and weaknesses (W) related to usability before and after conducting usability evaluation

Student	F1				Variance	
	1 DC		2 DC			
	S	W	S	W	S	W
A1	1	2	1	6	0	+4
A2	1	1	0	6	-1	+5
A3	1	2	2	6	1	+4
A4	3	4	0	4	-3	0
B1	2	1	1	4	-1	+3
B2	4	0	3	3	-1	+3
B3	2	1	2	3	0	+2
B4	1	1	1	2	0	+1
B5	1	0	1	3	0	+3
Total	16	12	11	37	-5	+25

The change in the students' opinions between the 1DC and the 2DC, can be grouped into three categories: reduction in the number of strengths and an increase in the number of weaknesses (we identified this category as 'expected change'), no change in the number of strengths and weaknesses (identified as 'no change') and increase in the number of strengths (we identified this category as 'unexpected change').

In the first case, an increase of weaknesses and a reduction of strengths related to usability, present a clear pattern in the change of opinion. After the evaluation, the students changed their opinions in order to report more weaknesses and a lower level of strengths related to usability in their software. The most representative change was given in the high number of weaknesses related to usability reported after the evaluation. For instance, in 1DC the student A-2 provided only one weakness related to usability: "looks awful", although in 2DC, the same student provided six new ones, e.g. "some counterintuitive stuff", "not enough buttons in specific windows", "same labels names – different actions", "confusing interface", "not enough label information" and "not enough indication of selected stuff". Other students also changed their opinions in an important way. This was the case for student A-3 who provided 2 weaknesses in 1DC, but after the usability evaluation, gave 6 weaknesses, e.g. "not consistent in all menus", "dropdown menu blocks buttons", "search function hard to find", "button names can be misleading", "some buttons are missing" and "windows too small". In 2DC, the same student also repeated this last weakness ("windows too small"). A lower variation in weaknesses was presented when it came to the change of opinion of student B-3. First, during 1DC, this student gave only one weakness: "slow UI between normal & full screen". Following this, in 2DC, the student provided three new weaknesses, e.g. "the learning curve", "full screen design flawed" and "bad keyboard navigation".

There were some cases where the students did not change the number of strength and weaknesses related to usability in their software. For example, student B-3 provided two strengths during 1DC such as "non-distracting design" and "intuitive design". In 2DC this student seemed to maintain his emphasis on the design matter; at that moment he reported two strengths, e.g. "smooth playback" and "nice design in normal mode (not full screen)".

Finally, there was an unexpected change in strengths. Student A-3 provided an additional strength after 2DC. In 1DC this student provided only one strength related to usability: "detailed overview for each entry". In 2DC the student maintained the same strength and gave another: "easy to learn". This student has broken the pattern related to reducing the strengths and increasing the weaknesses associated with usability.

3.3 Detailed Results on the Relative Importance of Usability

Our study also collected data relating to the relative importance which the students gave to software and usability matters, before and after their conduction of usability evaluation. These data were collected using the form designed to measure the relative importance given by the students to software/usability concepts (form coded as F2).

These results allow us to see the change in the understanding of usability from another perspective. Our interest was to identify whether or not the students placed more importance on usability matters after conducting the usability evaluation, and if there was a change, how this change occurred.

In this part of our study, we identified two main changes. The first change occurred when the students changed their opinion in order to prefer more usability matters. This change was coded as ‘X->U’. On the other hand, the second change occurred when the students had selected software matters such as more important. This alternative change was coded as ‘X->S’. Finally, our study also identified one case where no change occurred. We triangulated these results with the students' opinions related to their strengths and weaknesses of their software in order to verify consistency in the results. In Table 4 we present details of these changes.

Table 4. Detailed changes in the relative importance given by the students to software/usability matters, after conducting usability evaluation. (P# Pair of concepts)

Student	P1	P2	P3	P4	P5
A1					
A2	X->U		X->U		
A3			X->U		X->S
A4					X->S
B1	X->U				
B2	X->U				
B3			X->U		
B4	X->U			X->U	
B5		X->S		X->U	X->U

After conducting usability evaluation the students changed their opinion with the aim of considering usability matters as more important. These changes were particularly evident in Group B (students of computer science). Conversely, the group with more change of opinions towards technical aspects of the software, was Group A (students of software engineering). Finally, the common changes of opinion made to place more importance on the usability matter, were oriented to aspects related to designing GUIs and how to apply paradigms which could help this design.

3.4 Patterns in the Understanding of Usability

After identifying and understanding the improvement in the students’ understanding of usability, we focused on exploring whether or not it would be possible to identify the detailed characteristics of this improvement process. In order to systematize the identification of the patterns presented in this process, we proceeded to classify the opinions given by students in both 1DC and 2DC. We focused on those opinions which were related to usability, ignoring the opinions coded as technical aspects related to software. Here, both the strengths and the weaknesses are treated together as a unified group of opinions; our interest was to identify the characteristics of the opinions in general, regardless of their nature. The approach of taxonomy of usability proposed in [1], provided us with the framework for the classification. This taxonomy defined six attributes presented in the concept of usability: Knowability (K), Operability (O), Efficiency (E), Robustness (R), Safety (S) and Subjective satisfaction (SS).

In the case of Group A, the opinions are related to the attributes which are more oriented to users (K, O and SS). It is remarkable that the emphasis from students is placed on aspects connected to the "knowability" attribute, especially after the usability evaluation. The "knowability" attribute is defined as "the property by means of which the user can understand, learn, and remember how to use the system" [1]. For example, two weaknesses reported by the students were "Some counterintuitive stuff" and "Not enough indication of selected stuff". In the same way, one of the strengths was "Easy to learn".

This apparent concern of students for the user needs seems to be produced after the usability evaluation rather than at the same time. During the interviews that we held with two members of this group, their opinions seemed not to show a special affinity by the user during the evaluation. When we asked the students what they were thinking when they saw the users during the tests, one student said "... it can be quite funny to see users operate your program, especially when you make some easy task like finding a button, something that they may find difficult because your program may have some design issues". Another student, reflecting on a specific mistake that all the users found, reported that he "felt embarrassed because in the case of the mistake, it was an obvious mistake, never mind that the users found others mistakes too." More specifically, when we inquired about some special feeling of students toward the users during the tests, the first student responded: "Not really, just found it a bit hilarious, because our design was flawed". The second student reported: "I don't remember to have any specific feeling for the users; I just tried to be as objective as I could. I just focused taking notes all the time".

Next, we showed the students the information provided by them during 1DC and 2DC. We also showed them the change presented in their opinions between those DCs. At this time we asked them if they had realized, at the time of the 2DC, that their change of opinion was more oriented to usability. The first student stated: "Not sure if I was aware of it or not. Might have been since we've put a decent amount of effort in correcting our design mistakes afterwards". The other student reported that "Yes, I thought that I was more usability oriented, when I filled this form because I had my eyes open for the usability part of our software. I really notice which things the users felt using our software". Finally, we wanted to know if the students thought that their feelings toward users had been changed after observing the usability evaluations; their answers were categorical. The first student stated "Well yes, I did not take the user into account before, well of course a little bit but not as much. Lesson learned overall, that the user knows how the users want the design, the designer does not". The second one said "Well, I felt thanked for the users for point out the mistakes we made in our software". These partial results confirm that the students recognize the importance of users, that they express a genuine interest in those usability issues more connected to users' needs, and finally, that these feelings seem to be generated after the evaluation.

On the other hand, in the case of Group B again here it is possible to see a clear orientation to “knowability” attributes, e.g. the weakness “Relevant help information on every form” and the strength “Buttons have size compared to how often they are used”. In addition, these students also chose opinions related with the attribute “operability”, defined at the taxonomy as "the capacity of the system to provide users with the necessary functionalities and to permit users with different needs to adapt and use the system”. For example, one of the weaknesses was “The learning curve” whilst one of the strengths was “Easy to use when have been used once”. Finally, the students also selected opinions connected to the attribute “Subjective satisfaction” (e.i. “the capacity of the system to produce feelings of pleasure and interest in users”). In this case, one of the weaknesses and one of the strengths reported by students was, respectively, “Could have had a prettier GUI” and “It looks nice”.

Contrary to the previous group, the students of the Group B distributed their opinions in those attributes more oriented to users (K, O and SS). It could be possible to explain this difference based on the conditions in which students of Group B made their usability evaluation. These students worked with more users who developed more tasks, something that allowed these students find more usability problems.

We also held an interview with one student of this group in order to try to identify when this affinity by users’ needs occurred. The results were quite similar to those obtained in the previous interviews.

In general, all the students’ opinions show two characteristics. First, their opinions are oriented toward usability attributes and fully oriented to users’ needs. Second, after conducting usability evaluations, this phenomenon increases, specifically with regards to the concern of the students for aspects related to the needs of the users when it comes to understanding, learning, and remembering how to use the software. In Figure 1 we present these results.

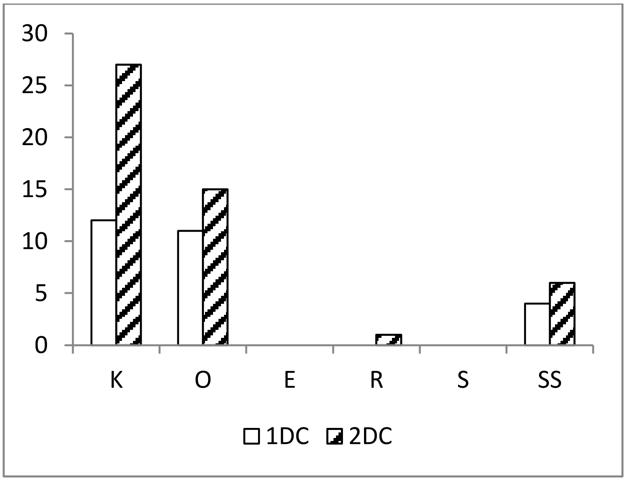

Fig. 1. Distribution of all students’ opinions regarding the usability of their software, before and after conducting usability evaluation

4 Discussion

The lack of understanding regarding usability is a factor which limits the application of usability activities as, for example the application of usability evaluations in software development [2, 3]. In our study, the lack of understanding regarding usability is represented in those initial perceptions of students about usability in their software. Before the usability evaluation, this perception was characterized by a lower number of weaknesses related to usability (32%). However, after the usability evaluation, developers changed dramatically this opinion and have reported numerous weaknesses (77%). There is also a change in strengths after the usability evaluation. This initial measurement of the status of the students' understanding of usability, is another example of the low level of relevance that developers normally give to usability matters due, among other things, to their different aims, motivations, or mindset [3], [11], [22]. However, after conducting the usability evaluation, the new measurement of the status of the understanding allowed us to identify a new different perspective held by the students. The corrective action used (usability evaluation) allowed students to gain a different perspective of their software: the users' perspective. At that moment, they could identify new problems in their software (i.e. usability problems or even other functional problems). More important is the fact that their perspective changed in order to realize the relevance of other usability matters. Additional evidence of this change in students' perspective is presented in the analysis of the importance which students gave to usability matters. After conducting the usability evaluation the students changed their opinion, placing more importance on usability matters. These changes were particularly evident in Group B (students of computer science).

This increase in the students' understanding of usability is connected to their empathy toward users, which was increased during the usability evaluation [10]. This is something that we also found in our study when we saw students focusing more on usability issues, after the usability evaluation. This general predilection for usability more than for other technical issues, allows us to infer more attention on users' needs. In addition, analysing the pattern in the understanding of usability allows us to identify that the students certainly had, but more important yet, have increased their attention to usability matters which are strictly connected with users' needs (i.e. knowability, operability, and subjective satisfaction).

Some students (Group A) emphasized their opinions in the knowability attribute. Others spread their opinions on all the attributes connected with users' needs. This could be explained in the characteristics of each usability evaluation. Students of Group B interacted with more users who made more tasks; more usability problems were found during this process. These students worked more time with the users consequently, this higher level of interaction with them allowed students to have a wider vision of users' needs.

The reinforcement in the pattern of the understanding about usability, generated after conducting usability evaluation, suggest some affectation of the students as a result of the observation of the users interacting with their software. This does not occur simultaneously at the same time as the interaction with the users.

The interviews with some of the students clearly allowed us to identify that during the moment of the evaluation, they were not focused on the users. Their concerns at that moment were more of a personal nature. This is the case for one student, who was in charge of conducting the evaluation and expressed his concern because the users had problems thinking out loud. Other student found it funny that users could not use the software system well due to some design flaws, or finally the case of the user who felt embarrassed. All these feelings are strictly personal. Furthermore, all students were conclusive in affirming that during the process they had no special feeling toward the users. However, evidence of empathy is clear when we see the improvement in the students' understanding of usability and the pattern of this understanding. In actual fact, analysing the feelings of students toward users, at the moment of the interviews, we see only positive thoughts towards them.

This unconscious acquisition of empathy by students is crucial in order to gauge whether the process behind the generation of empathy of the students is the contagion of users' emotions that they experimented with in their interaction with the users. Indeed, this unconscious process is the cornerstone of basic conceptualization of the EC theory [8], [17]. In actual fact, our study confirmed that the students acquired the users' feelings or emotions before generating an emotional empathy and, later the cognitive empathy which is reflected in their opinions during the 2DC and the interviews. These opinions are an example of the eventual affective response identified by [5].

This is not trivial, nor is it an elaborate explanation of a process which may seem very logical. Identifying EC as the source of empathy of students, allows us to realize that there are corrective actions which are more effective than other traditional options (e.g. regular training), in order to improve the understanding of usability. This is the case with the observation or conduction of usability evaluations by software developers. In our experiment we detected a level of understanding about usability at 1DC obtained by students, mainly as a result of the training received, including topics related to HCI. After the usability evaluation, the understanding of usability changed radically. This new level of understanding, and empathy toward the users, was generated by EC as a result of the interaction with the users in more real conditions.

5 Conclusion

In this paper we presented the results of a quasi-experiment conducted in order to explore the origin of novice software developers' empathy toward users and its relation to the improvement process in understanding usability. We explored the status of the understanding of usability before and after a corrective action (conduction of a usability evaluation) made in order to enhance the understanding. The corrective action allowed the participants in our study to interact with users while they were working with a software system. In our study we explored in detail the improvement in understanding usability, in order to identify clues to help us trace the origin of the empathy toward users, produced as a result of this improvement process.

We found a clear enhancement in the understanding of usability after applying the corrective action; we detected a new student perspective when it came to their software and also about the relative importance that they gave to usability matters over other software technical aspects. This change in the students' perspective reflects an impact on what Sohaib & Khan, as well as Lee, have identified as the aims and motivations of developers which are normally present in their mindset. A better understanding of usability should involve a higher level of empathy toward users; something which we explored by studying the patterns presented in the understanding.

Patterns presented in the understanding regarding usability before and after the corrective action draw a picture and thus make it possible to find a clear and generalized preference for those usability attributes fully connected with users' needs, i.e. knowability, operability and subjective satisfaction.

More relevant for us was the confirmation that this empathy towards users was acquired in an unconscious process of contagion generated during the interaction with users; something which is consonant with EC theory.

Our study attempts to fill the gap in the literature by explaining the origin of novice software developers' empathy toward users. Additionally, our research suggests that in any corrective action to improve the understanding of usability, there is something behind the scenes. EC plays a relevant role in these processes. EC theory explains why those actions which involve more interaction with real users, in real conditions, could have better results that other more traditional actions, such as training.

Considering that our results could only be generalized to novice software developers, it is necessary to conduct more longitudinal studies in order to explore how EC interacts with other kinds of software developers.

Acknowledgments. The research behind this paper was partly financed by National University (Costa Rica), Ministery of Science and Technology – MICIT (Costa Rica), National Council for Scientific and Technological Research - CONICIT (Costa Rica), and the Danish Research Councils (grant number 09-065143).

References

1. Alonso-Ríos, D., Vázquez-García, A., Mosqueira-Rey, E., Moret-Bonillo, V.: Usability: a critical analysis and a taxonomy. International Journal of Human-Computer Interaction 26(1), 53–74 (2009)
2. Ardito, C., Buono, P., Caivano, D., Costabile, M.F., Lanzilotti, R., Bruun, A., Stage, J.: Usability Evaluation: A Survey of Software Development Organizations. In: Proceedings of 33rd International Conference on Software Engineering & Knowledge Engineering, Miami, FL, USA (2011)
3. Bak, J.O., Nguyen, K., Risgaard, P., Stage, J.: Obstacles to usability evaluation in practice: A survey of software development organizations. In: Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges, pp. 23–32. ACM (2008)
4. De Vignemont, F.: The co-consciousness hypothesis. Phenomenology and the Cognitive Sciences 3(1), 97–114 (2004)

5. Decety, J., Jackson, P.L.: A social-neuroscience perspective on empathy. Current Directions in Psychological Science 15(2), 54–58 (2006)
6. Gilmore, D.J., Velázquez, V.L.: Design in harmony with human life. In: CHI 2000 Extended Abstracts on Human Factors in Computing Systems, pp. 235–236. ACM (2000)
7. Grudin, J.: Obstacles to user involvement in software product development, with implications for CSCW. International Journal of Man-Machine Studies 34(3), 435–452 (1991)
8. Hatfield, E., Cacioppo, J.T., Rapson, R.L.: Emotional contagion. Cambridge Univ. Pr. (1994)
9. Hertel, G., Niedner, S., Herrmann, S.: Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel. Research Policy 32(7), 1159–1177 (2003)
10. Hoegh, R.T., Nielsen, C.M., Overgaard, M., Pedersen, M.B., Stage, J.: The impact of usability reports and user test observations on developers' understanding of usability data: An exploratory study. International Journal of Human-Computer Interaction 21(2), 173–196 (2006)
11. Lee, J.C.: Embracing agile development of usable software systems. In: CHI 2006 Extended Abstracts on Human Factors in Computing Systems, pp. 1767–1770. ACM (2006)
12. Newell, A.F., Morgan, M.E., Gregor, P., Carmichael, A.: Theatre as an intermediary between users and CHI designers. In: CHI 2006 Extended Abstracts on Human Factors in Computing Systems, pp. 111–116. ACM (2006)
13. Patton, J.: Hitting the target: adding interaction design to agile software development. In: OOPSLA 2002 Practitioners Reports, p. 1–ff. ACM (November 2002)
14. Rasch, R.H., Tosi, H.L.: Factors affecting software developers' performance: an integrated approach. MIS Quarterly, 395-413 (1992)
15. Rosenbaum, S., Rohn, J.A., Humburg, J.: A toolkit for strategic usability: results from workshops, panels, and surveys. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 337–344. ACM (2000)
16. Rubin, J., Chisnell, D.: Handbook of usability testing: how to plan, design and conduct effective tests. John Wiley & Sons (2008)
17. Schoenewolf, G.: Emotional contagion: Behavioral induction in individuals and groups. Modern Psychoanalysis 15(1), 49–61 (1990)
18. Seffah, A., Metzker, E.: The obstacles and myths of usability and software engineering. Communications of the ACM 47(12), 71–76 (2004)
19. Shadish, W.R., Clark, M.H., Steiner, P.M.: Can nonrandomized experiments yield accurate answers? A randomized experiment comparing random and nonrandom assignments. Journal of the American Statistical Association 103(484), 1334–1344 (2008)
20. Singer, T., Lamm, C.: The social neuroscience of empathy. Annals of the New York Academy of Sciences 1156(1), 81–96 (2009)
21. Skov, M.B., Stage, J.: Training software developers and designers to conduct usability evaluations. Behaviour & Information Technology 31(4), 425–435 (2012)
22. Sohaib, O., Khan, K.: Integrating usability engineering and agile software development: A literature review. In: 2010 International Conference on Computer Design and Applications (ICCDA), vol. 2, pp. V2–32. IEEE (2010)
23. Shadish, W.R., Cook, T.D., Campbell, D.T.: Experimental and quasi-experimental designs for generalized causal inference. Wadsworth Cengage learning (2002)