How Novice Software Engineers Apply User Interface Design Patterns: An Empirical Study

Luis Rivero and Tayana Conte
USES Research Group, Instituto de Computação, Universidade Federal do Amazonas
Manaus – Brazil
{luisrivero, tayana}@icomp.ufam.edu.br

Abstract— **Conveying a positive User eXperience (UX) is essential for the success of any application, as it affects users' intentions to use a product. Nevertheless, novice software engineers may have difficulties in finding an appropriate solution for UX problems due to their lack of knowledge in interface design. User interface design patterns are well-working solutions to user problems that occur repeatedly which can help new developers solve interface problems. Although several studies investigated the effectiveness of applying design patterns in real development projects, few studies evaluate the difficulties that novice software engineers face when applying user interface design patterns for the first time. To understand how novice software engineers apply design patterns in the correction of UX problems, we carried out an empirical study with four small novice development teams redesigning mobile applications interfaces. We collected data on factors affecting the applicability of the employed design patterns, through questionnaires. Our results suggest that novice software engineers find this type of patterns useful but require means to find and select an adequate pattern to solve a problem.**

Keywords-design patterns; interface design; user experience; usability; software quality; mobile application, empirical study

I. Introduction

User eXperience (UX) has emerged as a new way of understanding and studying the quality in use of interactive products [1]. It is a consequence of a user's internal state, the characteristics of the designed system and the context within which the interaction occurs [2]. When developing a mobile application, UX plays a central role as applications conveying a positive UX are more likely to be adopted rapidly and gain acceptance [3]. Consequently, to convey a positive experience, it is necessary that the mobile application provides functionality and easy means to access it [4].

User interface design patterns can be a cost-effective alternative for correcting UX problems, as they describe best-practice solutions to problems faced by users during their interaction with an application [5]. These patterns also contain a discussion of when it is appropriate to apply the proposed solutions and what the consequences may be [6]. However, even though several authors evaluated the use of design patterns [6][7][8], few evaluate how novice software engineers deal with them while designing an interface. A proper understanding of the applicability of these patterns by software engineers is necessary to evaluate their feasibility in industry.

Qualitative methods are an alternative to better comprehend the issues that need a more specific and detailed analysis. They are recommended when it is necessary to consider human behavior in a research or when we need to understand the factors that affect the object of study [9]. In this paper, we carried out an empirical study using qualitative methods to better understand how novice software engineers apply design patterns and what makes it easy or difficult to employ them during the redesign of a mobile application. To this end, four teams of novice software engineers suggested modifications in the interface of two real mobile applications, using interface design patterns. Here, we present our findings regarding such use in terms of the factors that affected the applicability of the user interface design patterns.

The remainder of this paper is organized as follows. Section II presents a brief review on user interface design patterns and studies regarding their feasibility. Then, Section III presents the steps for carrying out the empirical study, while Section IV presents the analysis of its results. Finally, Section V presents a comparison of our findings with the ones found in literature and Section VI presents our conclusions and future work.

II. Design Patterns for User Interface Design

User interface design patterns have been introduced as a medium to capture and represent solutions to users' problems when interacting with a software application [8]. When designing software applications that provide a positive UX, user interface design patterns are tools that can provide an alternative to abstract and reuse the essential details of successful and usable interfaces, providing a positive UX.

A. Patterns Proposals for User Interface Design

A number of generic pattern languages have been proposed for suggesting both specific interface layouts and interaction steps. For instance, Tidwell [10] proposed a Pattern Language intended to support high-quality interaction between a person and a software artifact, aimed at passive activities (e.g absorbing information with little or no interactivity) to hands-on activities (e.g. creation of other objects). Additionally, Welie [11] proposed a pattern library which contains best practices with examples and insights on their applicability.

Patterns that are proposed for the design of specific types of applications (e.g. web, mobile, games, others) can be more effective in guiding software engineers in the correction of UX problems [12]. Consequently, specific patterns have emerged as a means to facilitate the development of mobile applications meeting users' needs. For instance, some patterns have been proposed by software companies (e.g. Apple, Google,

DOI reference number: 10.18293/SEKE2016-122

Microsoft) to support developers in designing applications for specific operating systems, while other patterns have been proposed for mobile devices in general. Examples of each of these types of patterns can be found elsewhere [13][14].

B. Evaluation of User Interface Design Patterns

According to Koukouletsos et al. [8], the potential of patterns and patterns languages has been widely promoted, but there has been little empirical evidence about pattern use and its benefits for interface design. Among the studies evaluating the use of design patterns in such context, Chung et al. [6] evaluated the effectiveness of their proposed design patterns for the design of ubiquitous applications. Furthermore, Lanzilotti et al. [7] studied how a pattern-based inspection approach could assist novice software engineers in the evaluation of interfaces. Additionally, Koukouletsos et al. [8] verified the difference between applying design patterns and design guidelines.

Even though the above studies evaluated the use of design patterns, few have focused on both patterns for mobile applications interface design and their applicability by novice software engineers. For instance, in the study by Chung et al. [6] some of the participants were experienced software engineers with knowledge in UI and Web design. Also, Lanzilotti et al. [7] did not focus on the effect of applying design patterns for (re)designing an application. Additionally, the evaluation by Koukouletsos et al. [8] was not performed on mobile applications. Consequently, there is still need to gather information on how novice software engineers apply mobile patterns in the (re)design of mobile applications interfaces. Such information will be useful for identifying improvement opportunities in the application of design patterns by novice software engineers in the industry, and for improving the UX of the developed applications.

III. The Empirical Study

We conducted the study at a university in the city of Manaus (Brazil) during a class on software quality at a Computer Science course. Some of the topics from the class were: usability/UX and ways of identifying and correcting usability/UX problems. The study took place during 6 weeks out of the 4 month duration class.

A. Participants

At all, 12 software engineering students in their last semester of college were enrolled in the class and agreed to participate in the study. These software engineers had technical background (more than 3 years of experience studying and/or practicing in the area) on software development methodologies, but low or no experience in interface design.

B. Materials and Procedure

We employed a qualitative research methodology aiming to understand which factors contribute or make it difficult to apply design patterns in the redesign of a software interface by novice software engineers. The students were divided into four groups of three novice software engineers each, and each group participated in all of the following stages.

Stage 1 - Lectures and training: At this stage, the novice software engineers participated on regular classes on the methods that were to be applied. A specific training regarding usability and UX evaluations was prepared as part of the class. Also, this training contained examples of how design patterns had been applied to correct usability and UX problems.

Stage 2 - Usability evaluation on mobile applications in the market: At this stage, each group of novice software engineers chose between two mobile applications for android devices and carried out a first usability inspection using the Heuristic Evaluation, a method that suggests ten rules for identifying usability issues [15]. The evaluated applications were: (a) Manaus Bus[1], an application that supports users in finding the right bus to their destination; and (b) Manaus in Theaters[2], which helps users find out about the different movies on display in the theaters of the city. We focused on these applications as they are entertainment and information mobile apps[3], which require a high degree of usability to facilitate their use and enhance the UX of the users. Therefore, they could be a good example of how redesigning an application could improve its perceived quality in terms of UX [3]. After collecting all inspection reports, the identified problems were verified by a high experienced analyst (with more than 5 years of experience in usability and UX evaluations) to check which problems were real and which were false positives. There was a meeting with each team discussing the inspection results.

Stage 3 - Correction of the identified problems applying user interface design patterns: At this stage, the novice software engineers employed the design patterns for mobile applications [13][14] and proposed changes in the user interface. We made sure that the software engineers knew that they could apply the design patterns as they felt appropriate. This was done in order to understand what process they would use to redesign the application when applying the design patterns and the difficulties they faced during such application.

To collect the qualitative data, right after finishing stage 3, we applied a questionnaire containing open questions to: (a) assess the acceptance of the use of design patterns for the improvement of user interfaces by novice software engineers; and (b) identify advantages, constraints and improvement opportunities for improving the adoption of design patterns in the software industry by novice software engineers. Table I shows each of the asked questions and their purpose. For instance, Q2 asks the novice software engineer which patterns (s)he thought were the easiest and hardest to apply, which aims at making him/her think of the reasons that make a pattern easy or difficult to use, preparing him/her for the next questions.

C. Data Analysis

We analyzed the data obtained from the answers to the questionnaires using procedures from the Grounded Theory (GT) method. GT uses a set of systematic data collection and analysis procedures to generate, prepare, and validate theories on social phenomena or processes [16]. Although the purpose of GT is the construction of theories, a researcher may use only some of its procedures to analyze qualitative data.

According to Strauss and Corbin [16], the coding process, in which concepts (or codes) and categories are identified, can

[1] https://play.google.com/store/apps/details?id=com.manausemcartaz
[2] https://play.google.com/store/apps/details?id=com.onibus.manaus
[3] https://support.google.com/googleplay/android-developer/answer/113475

be divided into three stages: open, axial, and selective coding. During the open coding, the researcher carries out the breakdown, analysis, comparison, conceptualization, and categorization of the data. During the axial coding, we examine the relations between the categories. The relations between codes can be defined by the researcher him/herself. Finally, the selective coding realizes all the process refinements by identifying the core category with which all others are related.

In our analysis, we have carried out the open and axial coding analyzing the answers to the questionnaires, but have not elected a core category yet because GT suggests the circularity between the collection and analysis stages until the theoretical saturation is reached [16]. Therefore, we do not claim that we applied GT, but some of its procedures as we decided to postpone the selective coding phase.

TABLE I. QUESTIONS FROM THE FOLLOW-UP QUESTIONNAIRE AND THEIR PURPOSE WITHIN OUR STUDY

ID	Open Question	Purpose
Q1	Which were the steps that you followed in order to apply the design patterns?	To identify how novice software engineers apply the design patterns.
Q2	Which was the most difficult design pattern to apply? And which was the easiest to apply?	To make the novice software engineer think on the aspects that make it easy or difficult to apply a design pattern.
Q3	Please, describe the difficulties that you face when searching/understanding/applying the design patterns.	To make the novice software engineers list concrete aspects that made it difficult to apply the design patterns.
Q4	Please, describe the aspects that made it easy to search/understand/apply the design patterns.	To make the novice software engineers list concrete aspects that made it easy to apply the design patterns.
Q5	If you had to redesign an application again, would you consider using design patterns? Why or Why not?	To understand the overall opinion of novice software engineers regarding the use of design patterns for the correction of UX problems.
Q6	What would you do to facilitate the redesign using the design patterns?	To identify improvement opportunities in the use of design patterns for improving the UX of a software application.

IV. RESULTS

We only received the questionnaires from 10 participants even though all 12 were involved in all of the stages. Below, we present our results, referring to the novice software engineers as subjects using the code SXX, where XX is the number of the software engineers from 1 to 10. We have categorized our findings according to the categories that were created from the qualitative data. Table II shows the quotes that from which our finding emerged for each category.

The **Process** and **Steps** categories explain how the novice software engineers applied the design patterns. In our analysis, a process is a collection of steps, but some steps were highlighted as they were incorporated by the novice software engineers themselves. Some subjects described how they applied the design patterns and how they included further stages to facilitate their applications. In this sense, some subjects thoroughly described the exact process (see quote Q01) while others were more direct (see quote Q02). Despite the varying levels of detail in the description of the processes, we identified some specific steps that were incorporated by the subjects but were not suggested during the training. For instance, some subjects indicated that they searched for other information sources; not only other design pattern proposals but examples of how the design patterns had been applied in real applications to verify their suitability (see quotes Q03 and Q04). Also, some subjects indicated how they selected which problems would be corrected, as well as which pattern they would apply to correct it. In this sense, some of the criteria for prioritizing a problem was based on the difficulty on correcting a problem (see quote Q05) or based on the main functionalities of the app (see quote Q06). Regarding the selection of a pattern, the criteria were: patterns that solve the problem (see quote Q07), patterns that do not make the app difficult to use (see quote Q08) and patterns that are more related to the app's logic (see quote Q09).

Regarding the **Difficulties** and **Facilitators**, the subjects listed several factors that affected the use of the applied patterns. For instance, subjects indicated that applying the user interface design patterns was difficult when (see examples on quotes Q10 and Q11):

- More than one pattern was available and one had to decide which was better
- It was not possible to find a pattern that fit the app
- It was difficult to adapt the solution
- The software engineer had no experience
- The software engineer did not know how to apply the pattern
- The software engineer did not understand the pattern,
- There were few examples of how the pattern was applied

Additionally, the subjects indicated that applying the design patterns was easy when (see examples on quotes Q12 and Q13):

- The explanation of the pattern was simple
- There is training
- There are more examples
- The software engineers knows what (s)he wants to achieve
- There are images, videos and others explaining how to apply them
- The name of the pattern is the same as the problem that needs to be addressed
- One can ask an expert about them
- There are tools which allow support the application process (e.g. tools for searching patterns)
- There are heuristics that help/guide when choosing a pattern.

All subjects indicated that they would apply the design patterns if given the chance (see quote Q14). Among the **Reasons for Employing Design Patterns**, the subjects indicated that they thought the design patterns would guarantee that the application would not be redesigned without following standards (see quote Q15) and that the app would be more visually appealing and organized (see quote Q16). Other aspects that motivated the subjects to employ the design patterns were: providing confidence in the results of the redesign; that the patterns were structured, which made them easy to apply; that they provide guidance and support; and that they provide solutions to common problems.

Finally, regarding **Improvement Opportunities**, the subjects indicated that patterns could be more specific so they do not become too ambiguous (see quote Q17). Also, they suggested that they themselves study the patterns and the steps for applying them (see quote Q18). This indicates that perhaps, they thought that the training was not enough (or could be improved) for applying the design patterns. Nevertheless, we highlight that we intended to let the subjects apply the design patterns freely, so we could verify what difficulties they encountered and what they included (or would include) in their application process to make it more effective. An interesting finding was regarding the use of the documents we prepared for delivering the activities. In order to collect the data on the redesign process and its results, in Stage 3 (see Section III), we provided the subjects with a template of a report, which asked the reason for choosing a specific pattern. That report made some of the subjects become more selective of the design pattern that they would apply, as they had to describe the reasons for such choice (see quote Q19). Also, having identified the problems that they had to correct based on an inspection method facilitated identifying the patterns that could correct them, as these selected patterns should made the application meet the heuristic's principles (see quote Q20). This would suggest that having a document guiding the redesign process and a set of rules that need to be followed when deciding which design pattern to use could be introduced into the application process, facilitating the use of the design patterns by novice software engineers. New studies would be necessary to test these hypotheses and evaluate the extent of support that these documents and guidelines could provide for novice software engineers.

TABLE II. QUOTES FROM THE NOVICE SOFTWARE ENGINEERS SUPPORTING OUR FINDINGS WITHIN THE STUDY

ID	Category	Sub	Description
Q01	Process	S06	*We had to identify the problems in the application and we searched for other sources containing patterns and possible solutions to the identified problems. We applied the patterns that better solved the problem, following its guidance and we made changes in the interface.*
Q02	Process	S08	*According to the application we were going to redesign, we chose, as a team, the best option that would fit that application*
Q03	Steps	S03	*I also looked up additional content, besides the one provided in the android patterns.*
Q04	Steps	S09	*I verified examples in other applications.*
Q05	Steps	S10	*We sought what would be easier to correct in the app*
Q06	Criteria	S10	*We worked on the main functionalities of the manaus bus application*
Q07	Criteria	S06	*I applied the pattern that better solved the problem" - criteria*
Q08	Criteria	S05	*I also verified which pattern would be simple enough and would not bother the user or make it difficult to use the app*
Q09	Criteria	S04	*Understanding the logic of the app (how it navigates, shows content, others) seems to be the best way to choose a pattern*
Q10	Difficulties	S03	*There was some doubt when applying the navigation pattern, where we would associate the menu option with the help icon, since both were associated with the action bar*
Q11	Difficulties	S10	*We based on the patterns for mobile applications and we did not find a pattern for that problem*
Q12	Facilitators	S04	*Yes, because the source where we found the patterns was very simple and direct*
Q13	Facilitators	S02	*I believe the easiest pattern was the 'links' since there was a pattern with a name that was specific for that*
Q14	Intention to use	S07	*Yes, i would definitely use them, as they follow the company's standards*
Q15	Reason for employing	S05	*It helps the software engineer a lot because there is a rule that makes sure that the application is not redesigned without guidance*
Q16	Reason for employing	S07	*Yes, with the redesign we noticed that it helped a lot in the new layout and made the application much nicer*
Q17	Improvement	S01	*Some patterns were too repetitive, so they could be made less similar and unambiguous.*
Q18	Improvement	S02	*I would study more so i could understand them and apply them.*
Q19	Influencing Factors	S05	*The template of the document helped when carrying out the redesign… i wondered if choosing the specific pattern would correct the problem.*
Q20	Influencing Factors	S06	*The heuristics [from the heuristic evaluation] were important for identifying the problems, thus they were also the reference for choosing patterns that would not violate them.*

ID – Code for the quotation, Category – Category from applying GT procedures, Sub – Subject ID, Description – Literal quotation

V. COMPARISON WITH PREVIOUS RESEARCH

We compared our results to the ones found by other authors to verify similarities or discrepancies. Regarding the work by Lanzilotti et al. [7], we found out that some of the difficulties in applying patterns were corroborated such as the difficulty in matching the correct pattern to a problem. The authors, however, identified that this could make software engineers disregard the correction of some of the identified problems. In our case, some of the software novice engineers indicated that when they did not know which problem they had to correct first, they based their decision on the importance of correcting the problem to achieving user goals.

While Koukouletsos et al. [8] did not provide qualitative results, their quantitative analysis suggests that applying patterns can improve the performance of novice software engineers. In this sense, the answers from the questionnaires within our study suggest the same, and provide reasons for such improvement such as making the redesign process easier.

Finally, Chung et al. [6] managed to obtain feedback from a subset of the participants who were novice software engineers. In this sense, they identified that participants using design patterns experienced less difficulties and extensively used the patterns in generating ideas and finding solutions. They also found out that another advantage of using the patterns were their usefulness to convey ideas to others within the group. Although our subjects did not report this feature, they indicated that they searched for further patterns in order to find further ideas. Additionally, we need to investigate whether the use of a template for reporting the use of the design patterns may have positively influenced the results of the study, and how could it be used to improve the performance of novice software engineers.

VI. Conclusions and Future Work

This paper presented an empirical study in which we verified to what extent novice software engineers can apply design patterns in the redesign of user interfaces of mobile applications. In their answers to a questionnaire of their experience applying the user interface design patterns, the novice software engineers indicated that they would apply the patterns again if given the chance and indicated specific activities such as how to select the problems to be corrected and patterns to address them. However, they indicated that the lack of experience and knowledge on how to apply the design patterns would make it difficult to do so. As improvement opportunities, we identified that document templates and a set of rules could be useful for guiding the redesign process, making it easier.

One of the limitations of our study was the small sample size employed and that the subjects were senior-level undergraduate students. However, even with the small sample, the results from this study allowed us to test the applicability of the design patterns for user interface design with positive results. Additionally, students who do not have experience in industry may have similar skills as novice software engineers in the industry [17]. Thus, the feedback provided by the subjects was useful for identifying aspects that made it difficult to apply the design patterns and improvement opportunities.

There could have also been a threat to the validity of our results in terms of the redesigned applications and their representativeness. Although two mobile applications were redesigned and evaluated, as there are many other categories of mobile applications (Games, Health, News, others), we cannot state that our results apply to all mobile applications. Nonetheless, Manaus Bus and Manaus in Theaters are two real applications which can resemble a real development and UX evaluation scenario, thus the qualitative data can provide useful information in the needs and improvements of the applicability of design patterns in order to facilitate their use by novice software engineers.

We also have to consider the subjectivity of the data classification as a threat to the validity of our results in the qualitative analysis, since it was performed by the first author of this paper. However, we used GT procedures in order to mitigate this threat, given that it requires the entire analysis to be grounded in the data collected. Additionally, the analysis process was performed along with another researcher, to encourage a better validation of the interpretations through the mutual agreement of the researchers.

Finally, one last limitation is the instrument and measures applied in this study for assessing the novice software engineers' opinion towards the use of design patterns. However, we believe that applying questionnaires was more suitable than applying interviews due to time constrains.

Given that each qualitative study provides evidences and hypotheses that can be later tested using quantitative methods, we intent to evaluate how including the improvement opportunities identified in this study can facilitate the application process of design patterns by novice software engineers. Our intention is to provide means of evaluating UX features and facilitating the redesign process through the use of design patterns, so novice software engineers can improve the quality of their applications, improving their acceptance.

Acknowledgment

We thank scholarship granted by CNPq to the first author of this paper, the financial support granted by CAPES through PROAP program and the financial support granted by FAPEAM through processes numbers: 062.00600/2014; 062.00578/2014. Also, we thank all the students who participated in our study as novice software engineers and the users who tested the apps.

References

[1] J. Bargas-Avila, and K. Hornbæk, “Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience,” Proc. SIGCHI Conference on Human Factors in Computing Systems, 2011, pp. 2689-2698.

[2] M. Hassenzahl, and N. Tractinsky, “User experience-a research agenda,” Behaviour & information technology, 25(2), 2006, pp. 91-97.

[3] M. Ervasti, S. Dashti, J. Reilly, J. Bray, A. Bayen, and S. Glaser, “iShake: mobile phones as seismic sensors--user study findings,” Proc. 10th international Conference on Mobile and Ubiquitous Multimedia, 2011, pp. 43-52.

[4] M. Hassenzahl, “The thing and I: understanding the relationship between user and product,” Funology, Springer Netherlands, 2005, pp. 31-42.

[5] A. Seffah, “The evolution of design patterns in HCI: from pattern languages to pattern-oriented design,” Proc. 1st International Workshop on Pattern-Driven Engineering of Interactive Computing Systems, 2010, pp. 4-9.

[6] E. Chung, J. Hong, J. Lin, M. Prabaker, J. Landay, and A. Liu, “Development and evaluation of emerging design patterns for ubiquitous computing,” Proc. 5th conference on Designing interactive systems: processes, practices, methods, and techniques, 2004, pp. 233-242.

[7] R. Lanzilotti, C. Ardito, M. Costabile, and A. De Angeli, “Do patterns help novice evaluators? A comparative study,” International journal of human-computer studies, 69(1), 2011, pp. 52-69.

[8] K. Koukouletsos, B. Khazaei, A. Dearden, and D. Tseles, “Comparing patterns and guidelines in web design,” Proc. 1st International Scientific Conference eRA, 2006.

[9] C. Seaman, “Qualitative Methods,” In: Shull et al. (eds.), Guide to Advanced Empirical Software Engineering, Chapter 2, Springer, 2008.

[10] J. Tidwell, “Interaction Design Patterns,” Conference on Pattern Languages of Programming, 1998. Available at: http://www.mit.edu/~jtidwell/interaction_patterns.html

[11] M. Welie, “A Pattern Library for Interaction Design,” 2011. Available at: http://www.welie.com/patterns/index.html.

[12] E. Nilsson, “Design patterns for user interface for mobile applications,” Advances in Engineering Software, 40(12), 2009, pp. 1318-1328.

[13] Google. Android Design Patterns, 2016. https://www.google.com/design/spec/patterns

[14] T. Neil, “Mobile design pattern gallery: UI patterns for smartphone apps,” O'Reilly Media, 2014.

[15] J. Nielsen, “Heuristic evaluation,” Usability Inspection Methods, John Wiley & Sons, New York, 1994, pp. 25-62.

[16] A. Strauss, and J. Corbin, “Basics of qualitative research,” Newbury Park, CA: Sage. J., Vol. 15, 1990.

[17] J. Carver, L. Jaccheri, S. Morasca, and F. Shull, “Issues in Using Students in Empirical Studies in Software Engineering Education,” Proc. 9th International Symposium on Software Metrics, 2003, pp. 239 – 249.