EVALUATING THE USABILITY OF MOBILE APPLICATIONS WITHOUT AFFECTING THE USER AND THE USAGE CONTEXT

Bodo Igler
Professor for Software Engineering, RheinMain University of Applied Sciences
Department of Computer Science, Wiesbaden, Germany
E-mail: bodo.igler@hs-rm.de

Tobias Braumann
Graduate Research Assistant, RheinMain University of Applied Sciences
Department of Computer Science, Wiesbaden, Germany
E-mail: tobias.braumann@student.hs-rm.de

Stephan Böhm
Professor for Mobile Media, RheinMain University of Applied Sciences
Department of Media Management, Wiesbaden, Germany
E-mail: stephan.boehm@hs-rm.de

─Abstract ─

Reliable usability evaluation results are crucial for the success of mobile applications. A high level of usability implies reduced development costs, reduced maintenance costs and higher customer retention.

State-of-the-art usability evaluations produce distorted results for mobile applications, as they are mainly tailored for desktop applications and mostly take place in artificial environments: The test user is isolated from the normal usage context and subject to observation processes which significantly affect the user.

This paper presents work in progress and proposes an architecture for a usability evaluation framework which preserves the natural mobile usage context, reduces the interference introduced by observations and reduces the overall costs of the evaluation procedure.

Key Words: *Mobile Applications, Usability Testing, Non-Intrusive Evaluation, Software Development, Innovation Management, Field Tests*

JEL Classification: L82, L86, L91, L96, O31, C93

1. INTRODUCTION

End-user satisfaction is pivotal to the success or failure of applications in the highly competitive app markets. Goldmedia Custom Research (2011) observes that there are several determining factors for customer satisfaction of mobile applications and that usability is the most important among these. Furthermore, usability is a determining factor for the overall software development costs. A high level of usability implies eventually reduced development costs, higher level of customer satisfaction, higher level of customer retention, and less money spent on maintenance and user training (Bias,2005).

Coursaris et al. (2011) evaluated more than one hundred mobile usability studies from 2000 until 2010. They observe that most of these studies are based on laboratory tests and that "there is no usability evaluation framework that yet exists in the context of a mobile computing environment". They consider the development of such a framework "an important topic warranting investigation".

A major strength of laboratory tests is the isolation and rigorous examination of cause-effect relationships. However, laboratory environments can only partially simulate the normal usage context of mobile applications (natural motion, interruptions, multitasking and noise) (Tamminen,2004). As field tests often involve special equipment and significant human resource allocation (Sá,2008;Kjeldskov,2004), a major drawback of field tests are potentially high costs.

This paper proposes a usability evaluation framework for mobile Android OS applications. The framework is required to satisfy four main objectives:

1. Non-intrusive usability evaluations: The usability tests take place in the normal usage context, while observation effects are kept to a minimum.
2. Non-intrusive app integration: Existing Android OS applications can be connected to the framework with minimum or no change to the applications.
3. Cost efficiency: Comparatively low costs for the field tests and the evaluation procedures (e.g. no expensive additional hardware in the field, no time-consuming manual procedures).
4. Extensible post-processing: The framework provides flexible support for e.g. aggregation and visualization of the test results.

Part of this framework has already been implemented during the research project SMAT (**S**uccess Factors of **M**obile **A**pplication Design for Public **T**ransportation).

The remainder of this paper is organized as follows: Section 2 summarizes the background and related work on architecture evaluation methods and usability evaluation; Section 3 presents the proposed architecture of the evaluation framework and applies architecture evaluation methods to it; Section 4 concludes the paper.

2. BACKGROUND AND RELATED WORK

2.1. Architecture Evaluation

This paper considers the design of the evaluation framework in the view of design science research (DSR). According to (Gregor,2011) DSR focuses on the design of IT artifacts of practical value, while keeping particularly the economical and social relevance in mind. Relevance is established by evaluating the artifact with respect to its environment. The artifact to be evaluated in this paper is the design of the framework itself. According to the design level presented in this paper, the architecture of the framework has to be evaluated.

There are two popular architecture evaluation methods which complement each other. RUP (Rational Unified Process) proposes an architecture evaluation method ("robustness analysis") which is based on interaction sequences derived from use cases. This method focuses on the evaluation of the functional requirements (Kruchten,2004). ATAM (Architecture Trade-off Analysis Method) focuses on the non-functional requirements. It proposes a stimulus-response modeling of concrete scenarios which are specific to the examined non-functional requirements (Clements,2011). Both the RUP and ATAM architecture evaluation approaches are conceptually related to the idea of case studies.

2.2. Usability Evaluation

The intended use of the proposed usability evaluation framework is to perform usability evaluations of mobile applications. This section briefly summarizes the research results on usability evaluation of mobile applications with respect to the above stated objectives. The design of the framework's architecture has to take theses results into account.

Remote usability testing maintains the normal usage context for mobile applications by gathering usage data and user feedback in the field. Several remote usability testing approaches rely on special hardware or accompanying observers (Sá,2008;Kjeldskov,2004). Other approaches rely on the mobile device to log the usage data in the field. This ranges from simple event logging to more sophisticated methods like capturing of usage paths or audio-visual recording (Liang,2011;Ma,2013). The latter approaches are less cost-intensive, but imply other drawbacks like difficult interpretation of simple log files and high data volumes.

There are several approaches for automated, non-intrusive usage data acquisition and evaluation. Most of the logfile evaluation techniques which were originally proposed for web application rely on simple algorithms. Tullis (2002) for instance derive usability results from task completion data and task time data. Usage path recording goes beyond simple event logging evaluations. The usage paths acquired in the field can be for instance compared to theoretically optimal usage paths which were recorded in the laboratory. Several studies state that this technique (sequence comparison technique) is easy to automate and that it can identify usability problems not found in lab tests (Ma,2013;Vargas,2010).

Most smartphones now provide video and audio recording capabilities. This information can be later replayed and analyzed. Depending on the hardware performance even video data can be immediately analyzed on the mobile client. Eye tracking has become a standard technique for usability evaluations in the laboratory (Bulling,2010). Current approaches employ the front camera of the mobile device for eye tracking (Lissoboi,2012;Miluzzo,2010).

Fully automated evaluation of field test usage data is mostly confined to simple conclusions. The derivation of cause-effect relationships in complex situations has to be supported by post-processing tools. These aggregate a potentially huge amount of data to information suitable for human examination.

Questionnaires as part of the mobile usage acquisition process can be automatically evaluated and provide usually more relevant data than simple logging methods (Ryu,2005;Väätäjä,2010). Nevertheless, it has to be taken into consideration that the incorporation of questionnaires into the mobile evaluation process raises the level of intrusiveness.

3. USABILITY EVALUATION FRAMEWORK

The proposed architecture of the usability evaluation framework is introduced in section 3.1. Subsection 3.2 outlines an exemplary selection of the applied architecture evaluation scenarios.

3.1. Architecture of the Framework

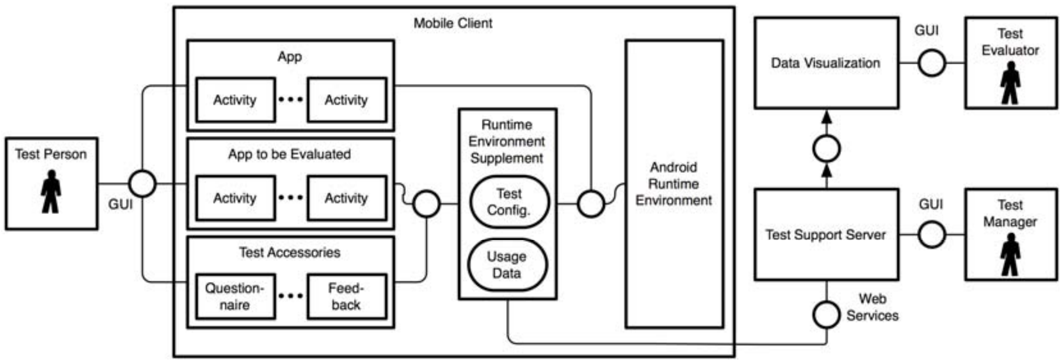
The proposed architecture of the usability evaluation framework (figure 1 and figure 2) is depicted in the form of FMC (Fundamental Modeling Concepts) block diagrams (Knöpfel,2006). The architecture comprises three main components (see figure 1):

1 Mobile clients: Apps whose usability has to be evaluated are embedded into the client-side of the framework.

2 Test support server: The test support server is used to configure and control all connected mobile clients. It stores and post-processes all usage data.

3 Data visualization: This component retrieves the processed usage data from the test support server. The data can then be further processed and displayed to the test evaluator.

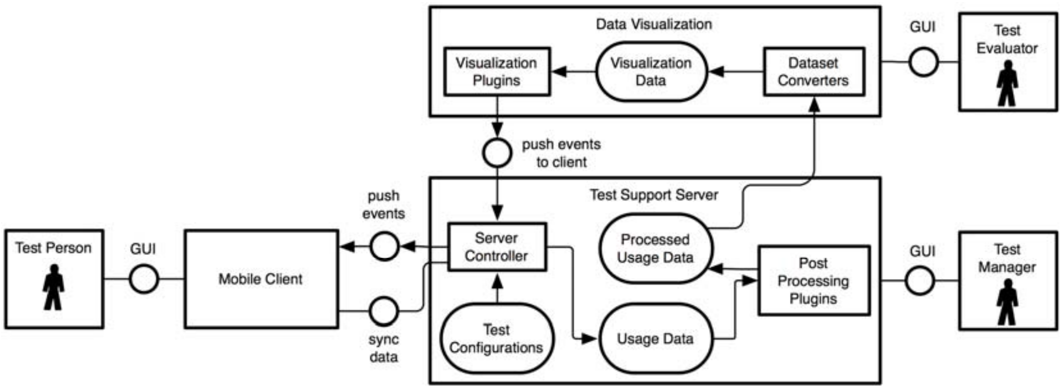
Figure-1: Framework Architecture with Focus on Mobile Client



Part of the client structure is predetermined by the Android architecture: Activities are the main components of each Android app. Each activity consists of one view and the corresponding functionality. A normal Android app is directly connected to the Android Runtime Environment. An app whose usability has to be evaluated is connected to the Runtime Environment Supplement. This supplement intercepts the communication between the app to be evaluated and the Android Runtime Environment and is thereby able to control the app to be evaluated based on the test configuration, and to acquire and temporarily store usage data. The test configuration and the usage data are synchronized via web services with the server anytime the mobile client is online. The other client-side component of the usability evaluation framework comprises the Test Accessories which are also controlled by the Runtime Environment Supplement.

A more detailed view of the server-side framework components is revealed by figure 2:

Figure-2: Framework Architecture with Focus on Server-Side



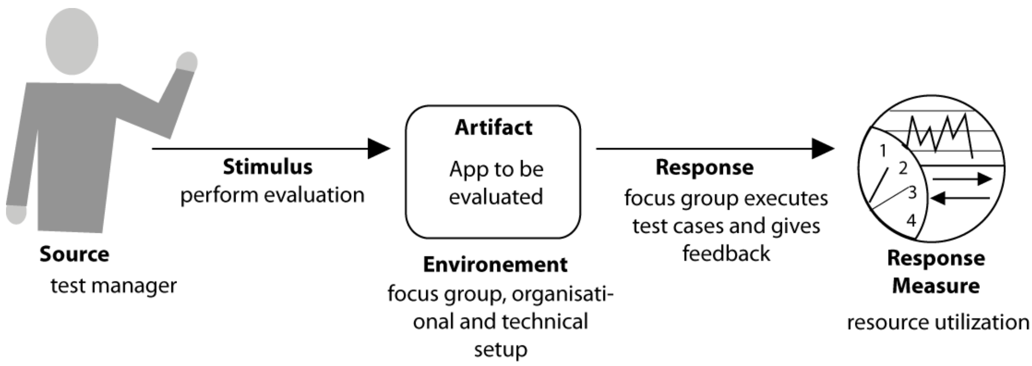
The test support server stores the usage data and the test configurations with all mobile clients in sync. The test manager is mainly responsible for the configuration of the tests and of the post-processing. The test support server provides the data visualization component with the processed usage data. It also accepts push events which are forwarded to the client. Push events can for instance trigger the feedback accessory in the client. The data visualization component supports the test evaluator by further processing and displaying usage data. The test evaluator can trigger activities on the mobile client. The data visualization component's architecture is based on (Malý,2011): Plug-in components support different types of data, processing and visualization.

3.2. Architecture Evaluation

The architecture of the framework has to be evaluated with respect to its functional and non-functional requirements. Robustness analysis is based on functional requirements in the form of use cases. From each use case several representative interaction sequences are derived. Interaction sequences give detailed accounts of the flow of information and control between all involved components of the architecture (including users as human components). Constructing these interaction sequences uncovers missing or unnecessary components, wrong communication links, ambiguities etc. Based on these findings the architecture is adapted and re-evaluated. The functional requirements for the framework proposed in this paper lead to both app-specific scenarios and scenarios which do not depend on the app to be evaluated. The app-specific scenarios are based on the use cases produced during the project SMAT. These use cases (e.g. “journey planner”) were developed for apps in the domain of public transportation.

The ATAM evaluation approach is based on the non-functional requirements in the form of quality attributes. In the context of this paper the most important quality attributes are: non-intrusiveness with respect to the usage context, non-intrusiveness with respect to changes to the app to be evaluated, cost-efficiency and extensibility. According to the ATAM approach stimulus-response scenarios have to be developed for each quality attribute. Figure 3 depicts an exemplary stimulus-response scenario which was developed during the project SMAT for the evaluation of cost-efficiency. The structure and graphical elements of figure 3 are based on the generic form for stimulus-response scenarios in (Clements,2011).

Figure-3: Scenario - Resource Utilization



ATAM scenarios are based on the idea of stimulating an information system such that the observation of the corresponding system response reveals quantifiable information on the quality attribute under consideration. In the example depicted in figure 3 the system is stimulated by the test manager who performs an evaluation. The system (artifact and environment) responds by executing the test cases and giving feedback. The system response has to be transformed into quantitative values. In this case the quantities for resource utilization (e.g. time and material for human resources and lab equipment) can be derived from the interaction sequences used for the robustness analysis above. To this purpose corresponding resource utilization data has to be calculated for each interaction item.

In order to decide if the architecture meets a specific quality attribute the response measure value has to be compared to a threshold value. The threshold values in this case have been derived by applying the same stimulus-response scenarios to a focus-group-based lab test not involving the framework. The comparison results indicate that the framework meets the cost-efficiency goal "less resource utilization than lab test without framework" for all considered interaction sequences.

Similarly scenarios were developed for non-intrusiveness with respect to the usage context. The scenarios for non-intrusiveness with respect to changes to the app to be evaluated and the scenarios for extensibility were obtained by applying the theory on modifiability scenarios from (Clements,2011). Again the considered interaction sequences indicate that the framework meets the objectives.

4. CONCLUSION

This paper reports on work in progress regarding the development of an integrated usability evaluation framework. The architecture of the framework has been designed. A robustness analysis has shown that it supports field test usability evaluations of mobile applications. Standard architecture evaluation scenarios have been used to examine the desired modifiability and extensibility properties. Specific architecture evaluation scenarios have been developed and applied with respect to the desired non-intrusiveness and cost-efficiency properties.

Part of the framework has been implemented during the project SMAT. This comprises in particular prototypes of the runtime environment supplement, of the feedback component and of the client-server data synchronization.

Further research will focus on the implementation of the framework, on the development of feedback and visualization components and on naturalistic evaluations of the framework.

ACKNOWLEDGEMENT

Part of this paper is based on findings of the research project SMAT, which has been partially funded by the HMWK (Hessen State Ministry of Higher Education, Research and the Arts) under the program *Forschung für die Praxis*.

BIBLIOGRAPHY

Bulling, A. and H. Gellersen (2010), “Toward Mobile Eye-Based Human-Computer Interaction”, *IEEE Pervasive Computing,* Vol.9, pp. 8-12.

Bias, R. G. and D. J. Mayhew (2005), Cost-justifying usability: An update for the Internet age, Morgan Kaufmann, USA:San Fransico.

Clements, P., R. Kazman and M. Klein (2011), Evaluating Software Architectures - Methods and Case Studies, Addison-Wesley:Amsterdam.

Coursaris, C. K. and D. J. Kim (2011), “A Meta-Analytical Review of Empiriacal Mobile Usability Studies”, *Journal of Usability Studies*, Vol. 6, No. 3, pp. 117-171.

Goldmedia Custom Research (2011), *Wichtigste Ansprüche der Nutzer an ihre Smartphones: Gute Bedienbarkeit und viele Apps,* http://www.goldmedia.com/presse/newsroom/mobile-monitor-2011.html, [Accessed 16.04.2013]

Gregor, S. and A. R. Hevner (2011), “Introduction to the special issue on design science”, *Inf. Syst. E-bus. Manag.*, Vol. 9, No.1, pp. 1–9.

Kjeldskov, J. and J. Stage (2004), “New techniques for usability evaluation of mobile systems”, *International Journal of Human-Computer Studies*, Vol. 60, No. 5–6, pp. 599–620.

Knöpfel, A., B. Groene and P. Tabeling (2006), Fundamental Modeling Concepts, West Sussex, UK: Wiley.

Kruchten, P. (2004), The rational unified process: an introduction, Addison-Wesley:Amsterdam.

Liang, H., H. Song, Y. Fu, X. Ca and Z. Zhang (2011), "A remote usability testing platform for mobile phones," (in: *IEEE International Conference on Computer Science and Automation Engineering (CSAE) 2011*, Vol.2), IEEE, pp. 312-316.

Lissoboi, I. and H. Kasai (2012), "Development of an Efficient Method for Eye Detection on Mobile CE Devices", (in: *International Symposium on Computer, Consumer and Control (IS3C), 2012*) , IEEE, pp. 337-340.

Ma, X., B. Yan, G. Chen, C. Zhang, K. Huang, J. Drury and L. Wang (2013), "Design and Implementation of a Toolkit for Usability Testing of Mobile Apps", *Mobile Networks and Applications*, Vol. 18, No. 1, pp. 81-97.

Malý, I., Z. Míkovec, J. Vystrčil, J. Franc and P. Slavík (2011) "An evaluation tool for research of user behavior in a realistic mobile environment", *Personal and Ubiquitous Computing*, pp. 1–12.

Miluzzo, E., T. Wang and A. T. Campbell (2010), "Eyephone: activating mobile phones with your eyes", (in: *Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds*), ACM, pp. 15-20.

Ryu, Y. S. (2005), "Development of usability questionnaires for electronic mobile products and decision making methods", *Doctoral dissertation*, Virginia Polytechnic Institute and State University.

Sá, M. de, L. Carriço, L. Duarte and T. Reis (2008), "A Mixed-Fidelity Prototyping Tool for Mobile Devices". (in: *Proceedings of the working conference on Advanced visual interfaces. AVI '08, Italy*), ACM:New York, pp. 225-232.

Tamminen, S., A. Oulasvirta, K. Toiskallio and A. Kankainen (2004), "Understanding mobile contexts", *Personal and Ubiquitous Computing*, Vol. 8, No. 2, pp. 135-143.

Tullis, T., S. Fleischman, M. McNulty, C. Cianchett and M. Bergel (2002), "An empirical comparison of lab and remote usability testing of web sites", *Usability Professionals Association Conference*.

Vargas, A., H. Weffers and H. V. da Rocha, (2010) "A Method for Remote and Semi-Automatic Usability Evaluation of Web-based Applications Through Users Behavior Analysis", (in: *Proceedings of the 7th International Conference on*

Methods and Techniques in Behavioral Research, Netherland), ACM:New York, pp. 19:1-19:5.

Väätäjä, H. and Roto V. (2010), "Mobile questionnaires for user experience evaluation", (in: *CHI '10 Extended Abstracts on Human Factors in Computing Systems*), ACM, pp. 3361-3366.