Mobile-Human Interaction Monitoring System

Ivan Pretel[1] and Ana Belen Lago[1]

[1] DeustoTech - Deusto Institute of Technology. 48007 Bilbao, Spain

Abstract. Information and Knowledge Society is involved into a more challenging phenomenon than ever. The ability of mobile devices to access information and services from anywhere and anytime is the main reason that empowers the massive usage of this kind of technology. Software quality has to be improved by developing mobile device interaction models according to the user necessities. These necessities have more kind of users than ever. It means the quality in use improvement is vital to the interaction. By this work we aim to give a more explicit point of view of the problems that appear during mobile application quality testing. In order to do so, we have studied a context model for mobile interaction design and the existing ways to capture, analyze and evaluate the user interaction. Finally we present one software solution consisted by a tiny mobile application and a desktop application. The exposed system can capture all necessary information to calculate quality in use metrics defined within ISO/IEC 9126 standard. The contribution revealed is a new approach to quality testing methodology focused on mobile applications where it is possible to improve the reliability of its results by paying special attention to minimize the influence of external elements used to monitor the interaction.

Key words: Quality in use, mobile services, context-awareness

1 Introduction

The Information and Knowledge Society has evolved into a more challenging phenomenon than ever. Information and communications technologies have been introduced into all fields of human activity linked by the key technology: full connectivity mobile devices. By this key, everybody can be aware of their economics and entertainment among others.

The ability to access information and services from anywhere is the main reason that empowers the massive usage of this kind of technology, which not only focuses on the individual but also on business and social groups. These social tendencies create the need for better awareness and readiness to face these demands quickly. The software quality is becoming increasingly important due to the exigencies of the new and aggressive mobile software market. In addition, the software design and development is progressively more focused on the user. Due to this tendency, a great amount of resources are invested in the long-term ambition of finding and developing mobile device interaction models according to the user necessities. In order to achieve this aim, knowing how the users feel using

the product and what kind of problems could have is vital. Inside the quality topic, ISO 9126 [1] standard describes the quality in use. This kind of quality measures how a product can satisfy the needs of the specified user to achieve specific goals in a particular context with effectiveness, productivity, safety and satisfaction. Unfortunately, a good software quality evaluation takes more time than the companies can invert.

The contribution revealed is a new approach to quality testing methodology focused on mobile applications where it is possible to improve the reliability of its results by paying special attention to minimize the influence of external elements used to monitor the interaction. Firstly, the quality focused on mobile devices interaction is explained in Chapter 2. Secondly, the context in use is studied and the context focused on mobile interactions is defined in Chapter 3. Capture methods and existing monitoring systems are studied in Chapter 4. The mobile interaction monitoring system is presented in Chapter 5. Finally, the research is concluded and further work discussed in Chapter 6.

2 Quality focused on the interaction

ISO 9126 defines a quality framework by three aspects: Intern Quality, Extern Quality and Quality in Use. Internal Quality is the totality of characteristics of the software product from an internal view (i.e. cyclomatic complexity, code maintainability). This kind of quality can be improved during code implementation, reviewing and testing. External Quality is the quality when software is executed, which is measured and evaluated focusing on the software application behavior (i.e. number of wrong expected reactions of software). Finally, Quality in Use is defined within ISO/IEC 9126-4. It is the quality of the software system that the user can perceive when it is used in an explicit context of use. It measures the extent to which users can complete their tasks in a particular environment. It is measured by four main capabilities of the software product in a specified context of use:

- "Effectiveness": The capability to enable users to achieve specified goals with accuracy and completeness.
- "Productivity": The capability to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved.
- "Safety": The capability to achieve acceptable levels of risk of harm to people, business, software, property or the surrounding environment.
- "Satisfaction": The capability to satisfy users.

These capabilities have to be measured in order to calculate how the quality in use of evaluated software is. Focusing on mobile devices, every software capability has to be measured per task and also per user, who is surrounded by the context in which actions are needed to be tracked. Owing to the wide range of contexts, an explicit context in use definition focused on mobile interactions has to be defined.

3 Context in Use

According to ISO 9241-11[2] standard, context in use is defined as every user, task, equipment and also physical and social environment that is affected by the interaction. In 2007, the NIST [3] institute published a new document adding every stakeholder to the context in use defined in the first standard. Other context in use definition is specified by Kankainen [4], he defines context in use as the environment that involves the user and his community. Nadav Savio and Jared Braiterman [5] explain the context by enumerating the following layers: culture, environment, activity, goals, attention, tasks, interface, device, connection and carrier. For mobile interaction, context is everything.

According to the exposed definitions, the different mobile context components are user, mobile device and environment.

The user has to be described by four main groups of attributes: personal, knowledge, skills and attitudes. Personal attributes are name, age and sex. The attributes related to knowledge are those attributes that can affect language, systems, products, work area, experience and eases with the tasks defined, culture, education level and experience using similar products. Physical abilities, mental abilities, disabilities and qualifications form the skills group. The attitudes group is formed by motivations, previous experiences and expectations.

The environment is also formed by groups of attributes: physical, ambient, technical and sociocultural groups. Inside physical group are attributes that describe the tangible environment (e.g. work area dimensions). The aim of ambient group is to keep attributes that can describe meteorological conditions, such as humidity, temperature or sound level. The sociocultural attributes group defines the cultural and social agents that can determine the user experience (e.g. cultural habits, religion). Technical group define every characteristic used during the tests excluding the mobile device, for example, connectivity attributes, hardware and software characteristics and so on.

If the work is focused on mobile environments studies, it appears the first complication. The main problem the quality in use shows is it is highly context-dependent. It is widely acknowledged that mobile environments are continuously changing. Therefore, context in use focused on mobile-human interaction (Figure 1) is formed by one mobile device, its owner, and also every environment that appears during the tasks execution.

4 Interaction data compilation

In order to define the best mobile-human interaction capturing we have studied the existing capture methods and the different advantages and disadvantages of the monitoring systems.

4.1 Capture methods classification

Firstly, we have studied the existing methods used to capture user interaction to conclude the best monitoring way. Different kinds of classification are found:

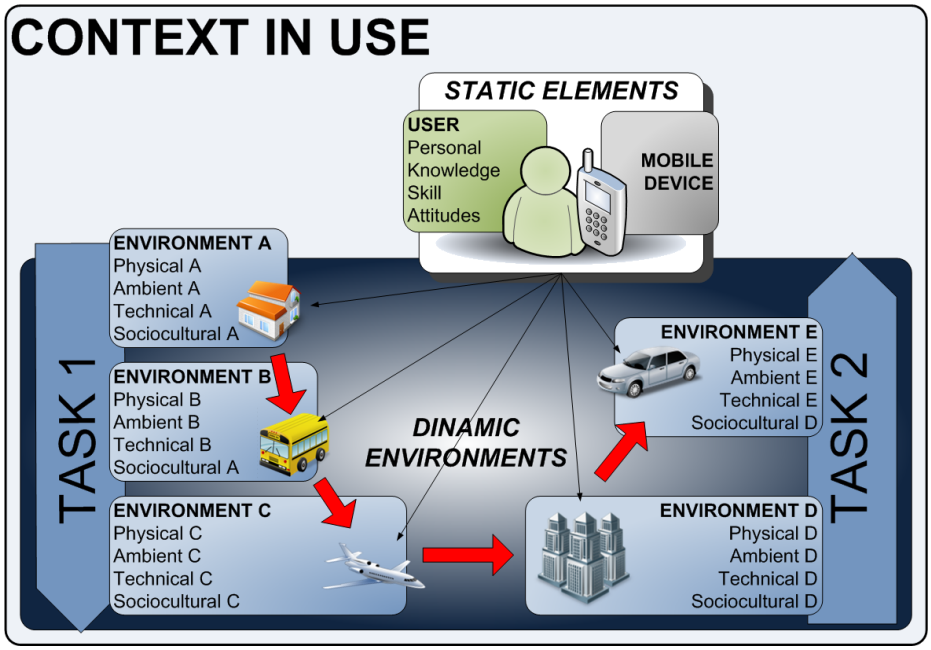

Fig. 1. Context in use definition

focused on environment, on automation level and also on the participation of users.

Focused on the environment which in the capture is developed, we can differentiate between real environments and laboratory environments. Testing executed in laboratory is easy because all influencing factors can be controlled and data can be recorded with several cameras and capturing tools. However, the context, which is the most influential factor, is not considered or it can hardly be simulated. In contrast to the laboratory methods, testing in real environment means that all data can be captured within real context influence.

Other kind of classification is by the automation level. They can be differentiated between manual and automatic methods. The manual method does not require any additional software or hardware and they are very flexible methods (it can measure specific and general parameters). The main problem this kind of method has is that it is extremely subjective. Because of that those captures could easily go wrong. However, automatic methods can capture highly objective information they are very quick.

The last method grouping is by the participation of users. Experiments done by real and direct interactions made by users (direct users and also stakeholders) can provide real data and can discover new problems. Contrary to the experiments with real users, experiments done by experts can only detect known problems and they also have more subjectivity than the first group.

After studying those methods, focusing on the objectivity and reliability of the captured information, the best method to capture user interaction is in real environments with real users in order to retrieve data as objective as possible and also the automatic methods to save time and add more objectivity to the capture. To sum up, we have explained the context in use is formed by user, mobile device and environment. Additionally, the best way to capture the most objective data is by monitoring the real user interaction in real environments by automatic systems. The next step is to study the best way to capture the information without influencing the context in use.

4.2 Existing monitoring systems

During the design of the system showed in this work, various interaction capture tools were studied: Morae [6], The Observer [7] and AQUA [8] among others. We have found a wide range of ways to capture the interaction.

The most common way is by camera installation in the device or added on a helmet. These added elements influence the context (the ergonomics of mobile device and the comfort of the user). When giving a sample, if a user whose phone has external capturing accessories (i.e. added camera), he will feel uncomfortable and he will change his behaviour. Consequently, this interaction will be corrupted and it will show worse quality results than without camera.

Interaction capturing by tests is a good method because they can be done after or before the interaction without influencing the context although they can add subjectivity depending on the design of the test questions.

Other way is by human observers. Although this method is manual, it has to be mentioned because is a good sample to understand the problem could lightly appear in camera installation and testing methods. The problem is that user can have a tendency to show expected (but not real) results due to the being observed and being evaluated feelings (i.e. if the user detects cameras, he will feel observed and evaluated). The most objective way to capture the interaction is by logs. The element that is altered is the mobile device because the logging software can reduce its performance.

In summary, if we capture data focusing on mobile context by the mobile device we can provide deeper and objective information without changing drastically the interaction. Therefore, the main goal to the capturer designed is to capture interaction data by registering information only using the mobile device.

5 Mobile interaction monitoring system

The exposed monitoring system is made up of a tiny mobile application that is able to capture the interaction and a desktop application that is able to simulate the interaction captured by the mobile application. It can capture data by saving screenshots and the user actions. Every key pressed is logged within its timestamp and its corresponding screenshot.

5.1 Methodology

In order to register the interaction, we have defined one methodology that is divided in five main steps.

- Firstly, the devices have to be configured. The configuration consists in defining tasks and contexts which in the users have to do the experiment. These tasks and contexts are defined by one xml file. This file is stored in phones to lent and ridden by the mobile application.
- Secondly, users have to borrow the configured phones and do the specified tasks. The task information is showed by the graphical user interface of the capturing system. When user chooses one task to do, he notifies to the application he is going to start. After ending the task the user notifies to the application the task is ended. The system stops capturing.
- When the user ends every task, he has to go the lent device back. The administrator has to dump the interaction data from the device to the system.
- After dumping information the system administrator has to introduce missing data (i.e. interaction errors, search times...) simulating the interaction by processing the recorded information.
- Finally, all necessary information to analyze the quality in use is stored. Therefore, the desktop application can generate graphs and reports.

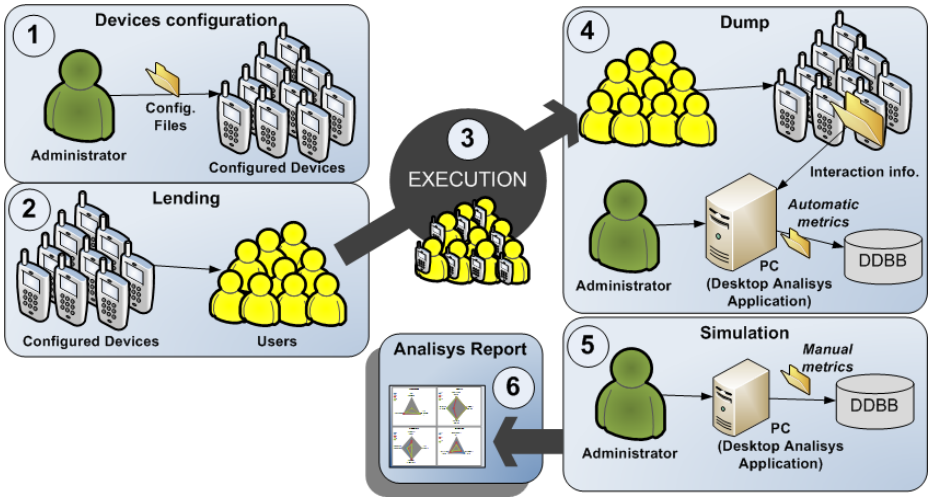

Fig. 2. Methodology

5.2 Architecture

The system is formed by two modules (see Figure 3): mobile application and desktop application. The first module is developed to be executed in mobile

devices. The files which contain the configuration, context and task information are ridden by the controller. It is developed in Java and its functionality is to interact with the user by its Graphical User Interface (GUI) and to send TCP commands (Start, stop, resume and pause) to the interface interaction capturer. The controller has been developed in J2ME language. Java Virtual Machine for J2ME has screen and key access limitation. If the application has not got the focus, it cannot access to the screen and the keys of the mobile device. This limitation was solved by developing interface interaction capturer (module that can capture data interaction) in PyS60 (Python on Symbian Series 60). This capturer can save information generated by the interaction in three types of files: image files (PNG format), test answers files (XML format) and log file (text format). Due to this module, the only element affected by the interaction monitoring system is mobile phone because its performance is reduced.

The second module is the desktop application. When users finish the experiment, desktop application takes this information and normalizes every data. This data is used to calculate metrics of characteristics that define quality in use. It stores this data in the database of the system (MySQL database). Another functionality this application has is simulation. It can simulate the interaction. By this way, we can capture the data that we cannot capture automatically. After seeing simulations and introducing incomplete data manually, it can calculate every metric and show analysis reports.

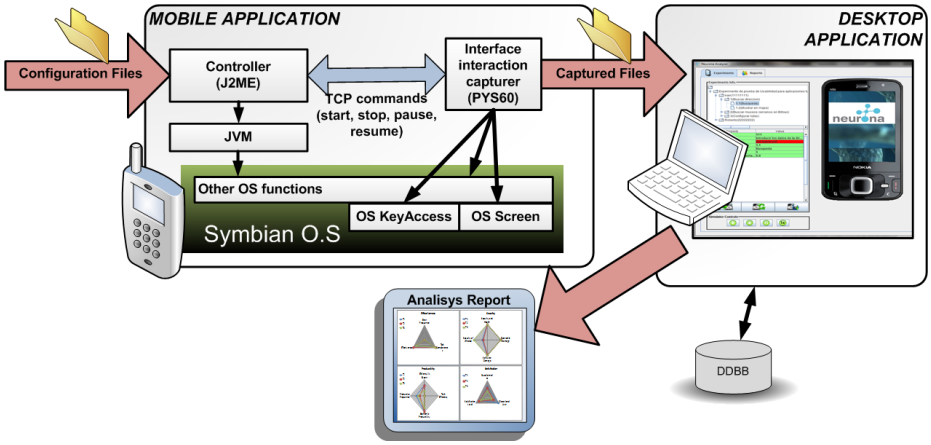

Fig. 3. Architecture

6 Conclusions and Future Work

The ability of mobile devices to access information and services from anywhere is the main reason that empowers the massive usage of this kind of technology.

Owing to this tendency, software quality has to be improved by developing mobile device interaction models according to the user necessities. These necessities that have to be satisfied by the new applications have more kinds of users. It means the quality in use is getting more important than ever. The exposed system can capture all necessary information to calculate quality in use metrics defined within ISO/IEC 9126 standard. By this work we aim to give a more explicit point of view of the problems that appear during mobile application quality testing. In order to do so, we have studied a context model for mobile interaction design and ways to capture the user interaction.

To sum up, this work reveals whether the quality in use testing is focused on mobile context, we have to take care choosing the interaction monitoring methodology because the context can be easily influenced. The solution presented by this work shows limitations and problems that have to be solved in future works. The first problem is we have to trust the user is doing the tasks in the specified context. Another problem is the device influence caused by the exposed capturing tool. These problems can be solved by automatic context detection and by optimizing the mobile capturing tool. This work concludes in order to design successful mobile interactions; we must be aware and understand the context in which they take place.

References

1. ISO/IEC 9126:2001. Information Technology - Software Product Evaluation Quality Characteristics and Guidelines for their use, 2001.
2. ISO 9241-11:1998(E). Ergonomic requirements for office work with visual display terminals (VDTs) Part 11: Guidance on usability, 1998.
3. NISTIR7432. Information Access Division Information Technology Laboratory, Common Industry Specification for Usability Requirements, 2007.
4. Kankainen, A. Thinking model and tools for understanding user experience related for information appliance product concepts. Dissertation of Anu Kankainen, 2002.
5. Savio, N, Braiterman, J. Design Sketch: The Context of Mobile Interaction: `http://www.giantant.com/publications/mobile_context_model.pdf` (retrieved on March 20, 2011)
6. Morae: usability testing and market research software: `http://www.techsmith.com/morae.asp` (retrieved on March 20, 2011)
7. The Observer XT: `http://www.noldus.com/human-behavior-research/products/the-observer-xt` (retrieved on March 20, 2011)
8. Covella G. J., Olsina L.A. Assessing quality in use in a consistent way. Proceedings ICWE '06, ACM, 2006, New York, NY, USA, 1-8