The analysis of contextual factors on the use of smartphones applications

A.H. Kronbauer[1],* and C.A.S. Santos[2]

[1] DMCC – UFBA / UNIFACS, R. Vieira Lopes n.2, Salvador – BA, Brazil
[2] DI – CT – UFES, Av. Fernando Ferrari s/n, CT VIII, Vitória – ES, Brazil

Abstract

The development of methodologies and techniques to evaluate smartphones usability is an emerging topic in the scientific community and triggers discussions about which methodology is most appropriate. The lack of consensus is due to the inherent difficulty on capturing context data in the scenarios where the experiments take place and on relating them to the results found. This work aims at correlate potential usability problems in mobile applications with contextual factors that may occur during users' interactions on different devices, such as environment luminosity, device screen resolution, and the user's activity while interacting with the application. The following methodology was applied to carry out a field experiment: (1) identification of contextual factors that may influence users' interaction; (2) use of UXEProject Infrastructure to support the automatic capture of applications' context data, by monitoring and storing quantitative, subjective and contextual data from applications' use; (3) implementation of experiments with real users, which have different profiles, using three different mobile applications over an one year period. In this paper, we present and discuss the results obtained during this study.

1. Introduction

With the continuous advance of wireless networks and the great proliferation of smartphones, many applications are launched in the market every day. Nowadays, the requirements for the ubiquitous computing, which predicts software as part of people's daily life and available, transparently, "at anytime, anywhere and from any device" [1], have been increasingly explored to build these applications. This application's ubiquity is reached by automatically monitoring contextual information related to the use of these applications. A context could be defined as a set of information that affects an application execution, related to people, objects, places, time and space in which the application is used [2].

Collecting data from smartphone users' experiences and associating them to the context where the interactions occur is a great challenge for the Human-Computer Interaction (HCI) area. The situations change and the results from the tests are highly dependent on the context. A person who is interacting with a mobile application while sitting on his home sofa, for instance, will have different external interferences when compared to the same task done while walking on the street.

As noted by J. Hansen [3], it is very important to relate the context influence on the users' interactions with mobile applications. To conduct the studies with such a broad reach, it is necessary to use methodologies and techniques that allow to carrying out experiments that are able to collect data contextualized with the scenarios where the interactions take place [4]. This fact provokes several discussions regarding the place where experiments are conducted (in the field or in the laboratory) [5], as well as the techniques

*Corresponding author. Email:arturhk@gmail.com

which may be used to extract the best set of data that characterize the experiments [6].

This work was motivated by these discussions, having the following main contributions:

- To identify the main evaluation approaches used to assess smartphones applications.
- To use the UXEproject infrastructure, as a new approach created with the potential to extract and relate quantitative, contextual and subjective data.
- To present the results of experiments conducted in the field, relating contextual factors to usability metrics for smartphone applications.

The remainder of this paper is divided into six sections. Section 2 presents the state of the art concerning evaluations of smartphone usability, encompassing the investigation of the approaches used to carry out the usability experiments. Section 3 introduces the UXEproject infrastructure adopted to facilitate the experiment presented in this work. Section 4 describes the methodology used for the execution of the experiment. Section 5 summarizes and discusses the experimental results obtained. Finally, conclusions and future prospects are presented in Section 6.

2. The State of the Art

The relationship between context and usability is an issue widely discussed by the scientific community which studies the influence of scenarios regarding the interaction with smartphones. According to Mallick [7], the experiences show that human beings usually interact with systems in unusual ways. Thus, user interaction tests conducted in real scenarios are essential to delineate the users' preferences and the consequent adaptation of products addressed to them [5].

Kawalek et al. [8] suggest evaluation methods that encompass different observation angles in the experiments done in this area, such as quantitative data (usability metrics), the subjective evaluation (users' feelings) and contextual data (for example, environmental conditions and the devices' characteristics). The main problem is the lack of literature covering approaches that support these three requirements combined in a single experiment. Generally, only one or two of them are related.

Coursaris and Kim [9] carried out a systematic data survey, from 2000 to 2010, which allowed them to identify that 47% of the works that evaluate mobile devices are done in the laboratory, 21% in the field, 10% used both scenarios and 22% are conducted without the participation of users. A point to be observed is that many studies don't consider the mobile feature of such devices, applying traditional evaluation methods. Another fact which calls attention in the results presented is that 47% of the studies evaluate individual and out of context tasks, 46% are based on the technology used and only 14% consider context variables and the users' characteristics.

Table 1. Works that investigate the usability of applications for smartphones

Authors	Technique Applied
Burigat et al (2008) [10]	Logging and Survey
Sodnik et al. (2008) [11]	Observation, Interview and Logging
Fitchett and Cockburn (2009) [12]	Observation and Interview
Chin and Salomaa (2009) [13]	Logging and Survey
Lai et al. (2009) [14]	Survey and Interview
Ebner et al. (2009) [15]	Thinking Aloud and Survey
Hansen and Ghinea (2009) [16]	Survey
Bødker et al. (2009) [17]	Survey, Focus Groups and Interview
Kim et al. (2010) [18]	Survey
Li and Yeh (2010) [19]	Survey
Maly et al. (2010) [20]	Logging, Thinking Aloud and Observation
Grønli et al. (2010) [21]	Survey
Kang et al. (2011) [22]	Survey, Technology Acceptance Model (TAM) and Structural Equation Model (SEM)
Fetaji et al. (2011) [23]	MLUAT, Survey and Heuristic Evaluations
Hegarty and Wusteman (2011) [24]	Survey and Think Aloud
Grønli et al. (2011) [25]	Survey
Sparkes et al. (2012) [26]	Survey, Think Aloud and Interview
Bradley et al. (2012) [27]	Survey
Schaub et al. (2012) [28]	Logging and Observation
Kirwan et al. (2012) [29]	Logging and Survey
Spyridonis et al. (2012) [30]	Survey and Interview
Silva et al. (2013) [31]	Survey, Observation and Interview
Spook et al. (2013) [32]	Survey
Ben-Zeev et al. (2013) [33]	Thinking Aloud
Othmana et al. (2013) [34]	Survey
Okazaki and Mendez (2013) [35]	TAM
Ferreira et al. (2014) [36]	Thinking Aloud and Survey
Reynoldson et al. (2014) [37]	Survey and Observation
Barros et al. (2014) [38]	Survey and Observation
Campbell et al. (2014) [39]	Survey
Molina et al. (2014) [40]	TAM and Tracker Device

With the objective being to identify the current reality of the usability investigations related to smartphones, a study was developed between the years of 2008 and 2014, encompassing works that describe empirical experiments

and investigate at least one of the following usability attributes: efficiency, effectiveness, satisfaction, learning, operability, accessibility, flexibility, usefulness and ease of use. The publications venues investigate were from the ACM, IEEE, Springer, and Google Scholar. Thirty-one works were selected, and they are listed in Table 1 along with the investigation techniques used.

The results of this study are detailed as follows:

- Concerning the techniques used for data collection in the experiments, it can be observed that 77.4% used surveys, 16.1% logging, 22.5% evaluators' direct observations, 22.5% interviews with users, 19.3% the Think Aloud technique, 9.6% used Technology Acceptance Model (TAM) and 12.9% other less traditional techniques. The sum of percentages exceeds 100% because 64.5% of the experiments encompass more than one technique, as shown in Table 1.
- Table 2 presents the number of times each attribute was incorporated in the investigations. It's possible to see that Ease of Use (100%), Satisfaction (94%) and Effectiveness (71%) are the most investigated attributes.

Table 2. Amount of times each usability attribute was investigated

Attributes	Number of times investigated
Ease of Use	31
Usefulness	16
Flexibility	7
Accessibility	6
Operability	3
Learning	13
Satisfaction	29
Effectiveness	22
Efficiency	17

- The number of participants was divided in three different ranges: 58% of the experiments used from 4 to 24 participants, 19.3% used from 25 and 44 participants and 22.7% were carried out with over 44 users.
- One of the main aspects to be highlighted is that only 3 experiments investigated contextual data, and were conducted in the laboratory, meeting the expectations and desires of a great number of researchers.
- The last issue to be pointed out is that none of the approaches captures the users' impressions concerning the application usability during their interactions, which could provide the correlation between the subjective data in the evaluations.

The main observation of the systematic study conducted was that in the majority of the experiments, surveys are used

to collect data, which might complicate the correlation between different kinds of information in order to find out usability problems [41]. Furthermore, in most cases, contextual factors are not investigated, an issue which is defended by many researchers as a primary factor for advances in the usability evaluations area [3][4][9].

In order to address some of those issues related to usability evaluation, next section presents the UXEProject infrastructure. This infrastructure was conceived to support quantitative, contextual and subjective extraction and correlation data to a better understanding of the real use behaviour of the analysed applications.

3. The UXEProject Infrastructure

The UXEProject infrastructure was built to give support to the usability evaluation based on the analysis of data captured directly from the devices. The formal model, which originated the infrastructure, can be found in full in [42].

The UXEProject infrastructure is conceptually divided in three units. These units comprise:

1. Mapping of tasks that will be investigated;
2. Combination of traceability metrics which enables the capture of contextual data, usability statistics and subjective information regarding the experiences provided to users and;
3. Storage and analysis of data captured during the experiments.

The infrastructure model uses a component-based architecture to support the reuse of the implemented resources and its redefinition according to evaluators' needs. Figure 1 presents the three high level components that represent the infrastructure and their relations: Mapping Unit, Traceability Unit and Assessment Unit. Arrows indicate information transfer between the components.
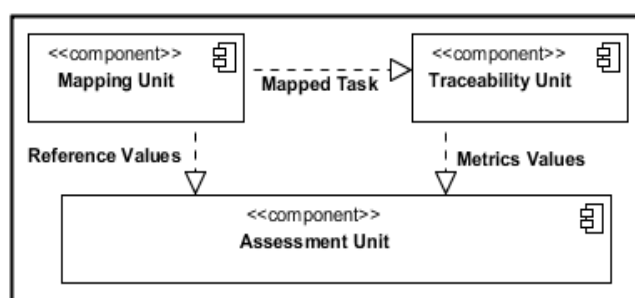

Figure 1. Main components of the infrastructure

To enable the automatic data capture concerning the user's interaction and the use of sensors present in smartphones, a Metric Library, developed with Aspect-Oriented Programming (AOP), is used.

In the infrastructure, the mapping of tasks is built through the capture of methods executed in the application that will

be evaluated. The Evaluation Team is responsible for the choice and mapping of tasks, as well as the creation of data capture metrics. It is important to emphasize that it is not necessary to have programming experience to carry out these activities.

The following subsections describe the infrastructure and tools used to encompass the predicted components in the three infrastructure units.

3.1. Mapping Unit

This unit is subdivided into three components responsible for providing the task-mapping functionalities. In Figure 2 the components diagram can be seen, with its interfaces and doors for data transfer from one component to another.
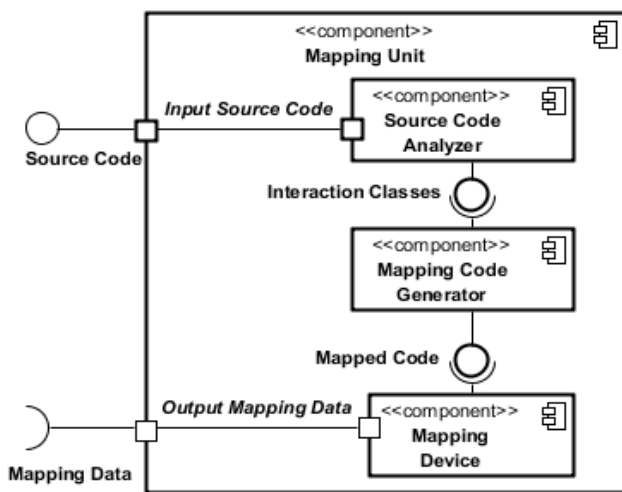

Figure 2. Mapping Unit Architectural Overview

Initially it has been planned that the applications' source code be made available. The objective of this action is to allow the Source Code Analyser component to identify which classes refer to the treatment of users' interactions.

These classes are identified and provided as a requirement for the Mapping Code Generator component so that it can attach information that enables the identification of tasks to the original source code.

The result of this component's action is to perform the instrumentation of the application's original code, thereby allowing the tasks to be mapped.

After the source code is prepared for mapping, it must be forwarded to the Mapping Device component where the tasks will be mapped. This device must enable the mapping of tasks and make them available to be used by other units in the model.

In Figure 3, the diagram portrays the sequence that enables the exchange of messages and data between the Mapping Unit components.

The first tool developed in the infrastructure encompasses the source code preparation to enable the mapping of tasks

provided in the applications. This tool was named Mapping Aspect Generator (MAG).

The MAG tool imports the source code from the application to be mapped and creates an Aspect that inserts the method onUserInteraction† in the classes that refer to the interaction layer. This process allows detecting the users' actions. In order to have the application ready to be mapped, it is necessary to compile the application source code with the Aspect generated. After that, it is enough to embed the application in a smartphone to make the interactions.

So that the Evaluation Team maps the tasks, another tool, named Automatic Task Description (ATD), was developed. The ATD should be embedded in a device and executed simultaneously with the application that will be mapped. Thus, as the Evaluation Team interacts with the application, the methods executed are automatically captured to be used as steps for the conclusion of a task.

The ATD method consists of the use of a filter that identifies when there is a user interaction. The filter identifies which classes, methods and parameters of the application were used. This information is stored in a XML file, which will be sent to the server to be used in the creation of metrics.
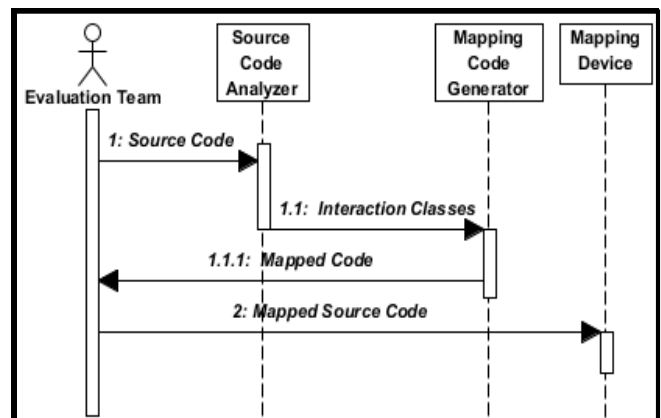

Figure 3. Mapping Unit Architectural Overview

3.2. Traceability Unit

This unit is responsible for collecting different types of information: (i) user profile (e.g. education and age); (ii) data referring to the user's interaction, such as hits and time to complete a task; (iii) contextual data like luminosity and noise; and (iv) subjective data, related to the users' feelings regarding the applications' usage.

For the instrumentation of the source code to include the code for capture metrics and data collection, the Traceability Unit was divided into three internal components, as can be seen in Figure 4.

---

† Further details at http://developer.android.com/reference/android/app/Activity.html#onUserInteraction()
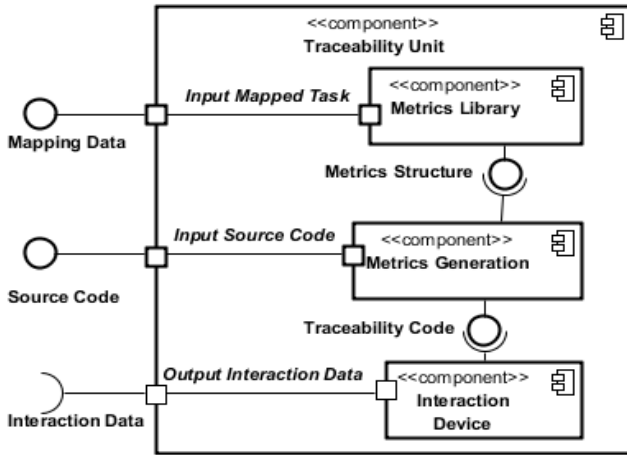
Figure 4. Traceability Unit Architectural Overview

Figure 5. Steps for the instrumentation of an application with the metrics for capturing data

The Metrics Library is the component that provides the structure to all capture metrics. It uses the mapped tasks to bring together the structure of the metrics with information about the application source code. It generates, as result, the metrics' structures adapted to the application under evaluation. In the development of the Metrics Library, the incorporation of three types of structure was expected: (i) the ones used to capture quantitative data from users interactions; (ii) the ones responsible for the subjective data (direct interactions with users) and; (iii) the metrics that allow instrumentation of the available sensors.

The Metrics Generation component receives as input the application's source code and the adapted metrics structure to be attached to this code. As result it provides a new source code with the inserted metrics, the Traceability Code.

The component Interaction Device requires the installation of the application's new code containing the metrics that will be used for data capture. This device should enable the user's interaction while allowing data to be captured.

In Figure 5, the diagram presents the sequence of actions that permits the exchange of messages and data between the Traceability Unit components.

The tool designed to allow the instrumentation of applications and to enable the data capture was named UXE Metrics Generation. This tool contains a library which has the structures of metrics to perform the measurements.

Initially, the tool has as input the XML file generated in the Mapping Unit. Then, the existing methods in the XML file are connected to the Metrics Library available in the tool, allowing the creation of Aspects responsible for the capture, transmission and persistence of data. At last, it is sufficient to compile the application's source code along with the Aspects generated and to embed the application in a device that will be utilized by a user.
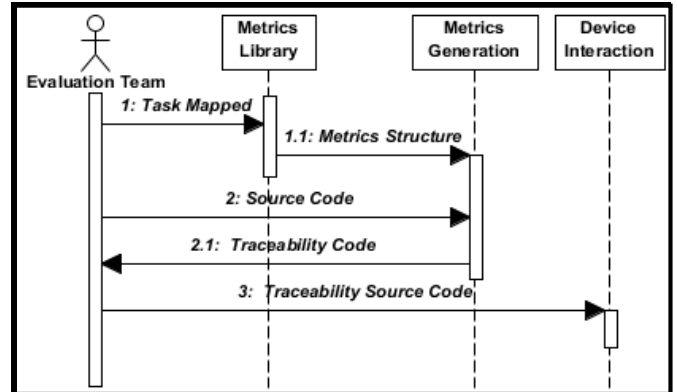
To encompass the data collection, three types of metrics were defined. The usability and context metrics use the Logging technique [10][11], and the subjective metrics use the Experience Sampling Method (ESM) [43] technique.

The efficiency and effectiveness of the users' interactions are measured through usability metrics described in Table 3. All the measurements take into consideration the mapping of a task, where all the steps to its conclusion are described. Thus, the errors, for instance, are identified every time the user interacts with the application that is not in the previous mapping.

Table 3. Usability Metrics

Usability Metrics	Objectives
Metrics for the capture of assertiveness	
Actions in conformity	Quantify the difficulty of interaction for the user in a task using the interface.
Actions in violation	
Metrics for the search and use of tasks	
Unfinished task	An unfinished task can mean that the user is lost or searching for a desired task.
Finished tasks	
Finished and used tasks	The use/non-use of a task indicates if the user found what s/he was searching for.
Finished and unused tasks	
Metrics related to performance	
Average time for task completion	Indicate the need for improvements that minimize the time and actions for the execution of tasks.
Average number of actions for task completion	
Number of tasks concluded without errors	
Metrics for the use of help	
Frequency of the use of help	Measure the level of difficulty faced by the user. Identify if the help available provides the desired effect.
Violations made after the use of help	

The sensors used to capture the contextual data were the accelerometer (to capture the horizontal or vertical position), GPS (to capture movement), Luminosity Sensor (to capture the environment's luminosity), and the microphone (to capture noise in the environments).

It is important to stress that the infrastructure uncovers new metrics to be incorporated in the Metrics Library, which increases its adoption in different scenarios, contexts, and with the use of other sensors. For example, specific metrics could be incorporated in an application for the spatial orientation of people. Therefore, such metrics could be associated with the data provided by a mobile device's GPS, enabling the comparison of the user's interactions with information regarding positioning, speed, and route taken. In that case, the usability analysis would take into account not only how a user performs a task but also how the environment influences the user's interactions.

The subjective metrics are used to measure the emotional state of the users during their experiences with an application. ESM [43] technique was chosen because of the following aspects: (i) it is appropriate for use on devices with relatively small screens; (ii) it is intuitive and doesn't take much mental effort to be interpreted; and (iii) it is capable of being answered with entry modes provided by different mobile devices.

ESM measures two dimensions, the kind of emotion (positive or negative) and the intensity of the emotion. To do this, a group of pictures is displayed indicating emotional states associated with a question, as can be seen in Figure 6. The sequence of the pictures represents varying degrees of emotional intensity and can be interpreted from left to right as: very displeased, displeased, indifferent, pleased, very pleased. These questions are defined by the evaluators during the execution of the Usability Metrics Generation Tool.
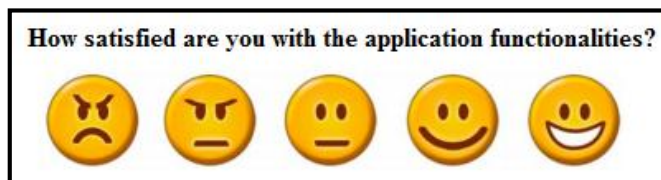

Figure 6. Example of ESM form used in the experiments.

In order that the data related to the experiments could be transmitted and stored on a database, a micro instance from the service known as Amazon EC2‡ was utilized.

3.3. Assessment Unit

This unit was structured in four different components that are responsible for infrastructure where data to be evaluated

‡ Available at http://aws.amazon.com/ec2/

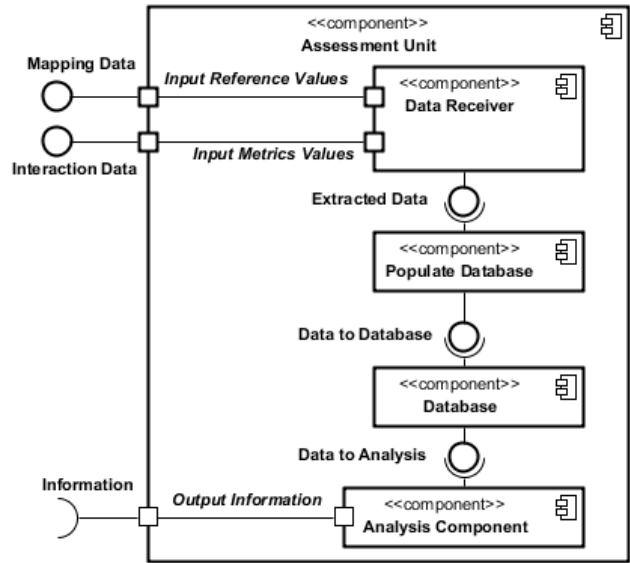will be stored over time. Figure 7 presents an architectural overview of its components.

Figure 7. Assessment Unit Architectural Overview

Initially, the Data Receiver component collects the data captured during the execution of the user's interactions within the application under evaluation. A second component, the Populate Database, performs the verification of new data arrival and populates it into the database. The Database component represents a Database Management System (DBMS) that stores persistent data. The Analysis Component has the responsibility to access existing data in the database and to provide resources that enable the extraction of usability information relevant to the evaluators.

To illustrate better its behaviour, the sequence diagram in Figure 8 presents the exchange of messages and data between the components.
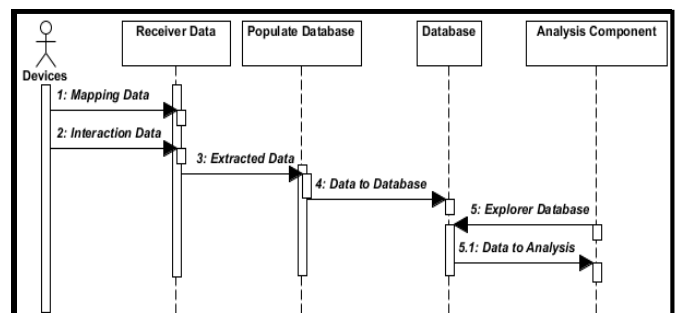

Figure 8. Actions performed by the components of the Assessment Unit

To encompass the components defined in the Assessment Unit, the following processes were performed: (i) create and

setup an FTP and a database (DB) server and make them available on the Internet; (ii) carry out the modelling of a DB and a Data Warehouse (DW) to store and enable the analysis of information captured during the experiments; (iii) create tools to detect the presence of new files in the FTP server, populate the DB and load the DW; e (iv) choose an OLAP tool to give support to the data analysis.

The Database Management System selected to store the data was the MySql Community Server§. In order to encompass the load of data on the DB, a tool named Data Load was developed. The steps executed by this tool are: detect the arrival of new files in the FTP server, extract the data and load them into the DB.

The last tool designed (ETL Maker) extracts, transforms and loads the data, transferring them from the DB to the DW. To make the data analysis easier, the OLAP tool Pentaho Analysis Services** was chosen.

The next section describes the case studies performed to evaluate the functionalities and identify the potentials of the UXEProject Infrastructure with the tools that give support to its implementation. The results of the experiment were presented in [44].

4. Experiment Conducted

The experiment reported in this article was divided in six different phases, based on the directives proposed in the DECIDE framework [45], which guided the specification of the steps during all phases of the experiment.

4.1. Determine the goals

The main focus of the experiment is to obtain information relating different kinds of data, with a major interest in the contextual factors which can interfere in the usability of the applications that will be analysed.

4.2. Explore the questions

Based on the objective to be reached, a set of questions was made to direct the experiments, and the data generation and analysis:

- How does the luminosity of the interaction's scenario interfere in the performance of smartphone application users?
- How does the user's movement interfere in his performance to interact with the applications?
- Which tasks are more affected by the position of the smartphone at the moment of the interactions?
- What is the difference in users' performance due to the smartphone setup?

---

§ Available at www.mysql.com
** Available at www.pentaho.com

- What kind of information can the context provide to improve the usability analysis?

4.3. Choose the evaluation paradigm and techniques

The evaluation approach to be used in this work should encompass the following conditions:

- The experiment has to be conducted in the field.
- Without supervision.
- For a long period of time.
- Data will be collected automatically.
- No restrictions concerning the number of users.
- No need to know how the applications were developed.
- Potential to be applied to any application for the Android platform.
- No need to have the Evaluation Team to write the programming codes.
- Possibility to analyse different kinds of data.
- Possibility to specify the tasks to be analysed.

In the face of the listed conditions, the UXEProject infrastructure was chosen, as it gives support to all requirements demanded.

4.4. Identify the practical issues

In this phase, a great number of prerequisites were raised, among which it can be highlighted: (i) the selection of applications to be evaluated; (ii) the definition of the investigated tasks; (iii) the definition of the group of users to participate in the experiment and; (iv) the data to be considered.

The first action in this phase was to conduct exploratory research aiming to find applications that have attractive functionalities and with the possibility to be inserted in people's daily life. The choice of applications considered the following prerequisites:

- The application must have been developed using the Java language and for the Android platform.
- The application source code must be available and have explicit rights of use.
- The application must have been built using good programming techniques, showing a good modularization of its functionalities, to allow the source code to be instrumented with AOP.

Description of Applications and Evaluated Tasks
The first application, called Mileage, aims to help the users to control their costs with fuel and other maintenance services of an automobile, such as oil change, brake pad change, among others. The left side of Figure 9 presents the application interface, and the right side, the tasks investigated in the experiment.

Mileage Interface	Instrumented Tasks
	1. Register a vehicle. 2. Input data of a new fill-up. 3. Setup a new data format. 4. Modify the data in the history of fill-ups. 5. Add a new maintenance control. 6. View the graph referring to the fuel price variation. 7. View the graph that refers to the travelled distance. 8. Import stored data.

Figure 9. Mileage Application Interface and instrumented tasks to be evaluated

The second application selected was ^3 (Cubed), a music and video clips manager. On its main menu, it is possible to select songs or videos and play them. Cubed interface and the tasks that have been mapped in this application are presented in Figure 10.

^3 (Cubed) Interface	Instrumented Tasks
	1. Choose a song from a playlist. 2. Create a new playlist. 3. Add a song to a playlist. Change the application's appearance. 4. Choose a new presentation theme for the application. 5. Open the Equalizer.

Figure 10. ^3 (Cubed) Application Interface and the instrumented tasks to be evaluated

The last application chosen for the experiment was Shuffle, whose interface is shown in Figure 11. It is an application which schedules the activities that allow to link

tasks to dates and times, besides permitting the association to projects and contexts (for example, home or work).
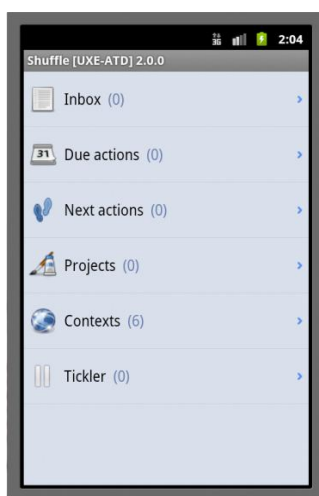
Shuffle Interface	Instrumented Tasks
	1. Add an activity. 2. Delete activities from the inbox. 3. Modify an activity. 4. Register a new project and choose an available context. 5. Register a new context. 6. Delete a context. 7. Modify a project by choosing a context. 8. Backup data. 9. Select the help option. 10. Create a scheduled activity

Figure 11. Shuffle application interface and instrumented tasks to be evaluated

Selection of participants

Another action completed in this phase was to define the group of users to participate in the experiments. The selection considered the profiles that were under analysis and their smartphones' features. Twenty-one participants were selected, taking into consideration the age, educational grade, education level, occupation and purchasing power.

Data Considered in the Experiment

The relationship of the data used in the experiment was defined according to the capture strategies provided by the UXEProject infrastructure. Thus, the usability data was considered related to the mapped tasks, the users' profile, the smartphones' features and the contextual data obtained through sensors.

The smartphones' features considered to compose the interactions context were the screen size and resolution, whose ranges of values are in Table 4.

Table 4. Smartphones Features

Features	Range of values considered	
Screen Resolution in Pixels	$\chi \leq 320 \times 240$	Low
	$320 \times 240 < \chi \leq 320 \times 480$	Medium
	$\chi > 320 \times 480$	High
Screen Size in Inches	$\chi \leq 2.4$	Small
	$2.4 < \chi \leq 3.5$	Medium
	$\chi > 3.5$	Large

So as to contextualize the environment where the interactions take place, the data are captured considering the degree of luminosity, the device position during the interactions and the speed in which the user moves. These context data are captured directly from the devices sensors and their reference values are described in Table 5.

Table 5. Scale of values for environment data

Factors	Range of values considered	
Luminosity	$\chi \leq 100$ lux	Low
	$100 < \chi \leq 10000$ lux	Medium
	$\chi > 10000$ lux	High
Movement	$\chi < 0.2$ m/s	Stationary
	0.2 m/s $\leq \chi \leq 2.7$ m/s	Walking
	$\chi > 2.7$ m/s	Motorized
Position	Vertical	Horizontal	Mixed

4.5. Decide how to deal with ethical issues

A site was built for the conduction of the experiment. It brings explanations concerning the research and a term of use of the applications. In order that the user is enabled to download the applications, it is necessary that he explicit his agreement on participating in the experiment.

4.6 Evaluate, Interpret, and present the data

The next section presents the results of the evaluations made during the experiment conduction. The data collection took place from 12/01/2011 to 11/30/2012.

5. Results of Experiment

5.1. Luminosity Influence Analysis

Initially, observed were the percentage values of tasks completed with errors in each application regarding the luminosity variation.
    The objective is to identify the possible influence of this contextual variable on the interactions. In order to conduct the analysis, the luminosity was isolated and related to the percentage of tasks completed with errors in each application, as presented in Figure 12.
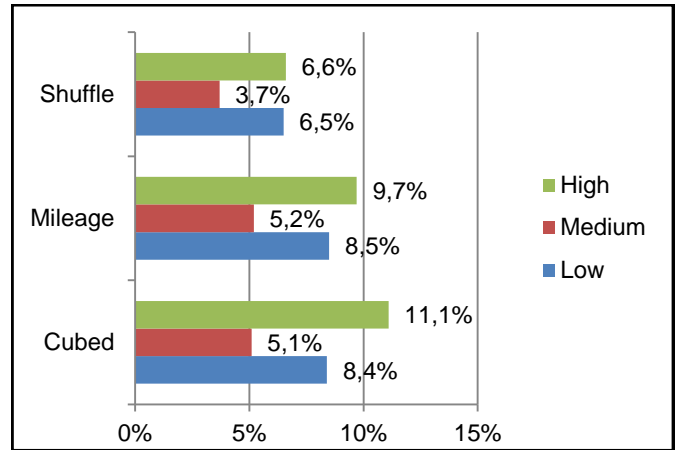
Figure 12. Errors rate due to luminosity

As we can see, it was detected that, for all applications, the highest rates of errors in completed tasks occur when the luminosity is either too high or too low, that is, when the interaction scenario's conditions are not within the parameters considered standard, which proves the luminosity influence on users' performance.

5.2. Movement Speed Analysis

The next evaluation refers to the speed in which users move when performing the interactions. The speed usually varies due to three possibilities: the user is either walking, or stationary, or in any kind of means of transportation (Figure 13).
    As seen in Figure 13, it is possible to identify in all applications that the actions performed with no movement show a lower error rate than the ones performed while moving.
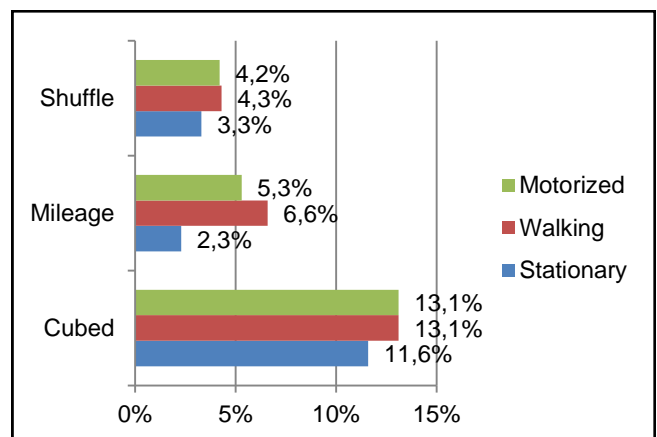
Figure 13. Error rate due to the movement speed

5.3. Position of Interaction Analysis

The second analysis regards the smartphone position during the interactions. The aim is to find usability problems in specific tasks related to the interaction position (vertical, horizontal or mixed). In Table 6, it is possible to see the tasks which had an error rate over 10% related to the position of interaction. This sort of information is useful for the application developers, as in future versions of the applications, the interactions in positions of high error rate can be inhibited. Table 6 shows that more than 50% of the problems occur when the tasks are done in a mixed position, that is, they are started in a position and ended in another.

Table 6. Error/failure rate due to the position of interaction

Position	Tasks – Mileage	% Errors
Vertical	Register a vehicle	16.7%
Mixed	Register a vehicle	14.3%
Vertical	Import stored data	21.1%
Horizontal	Import stored data	18.2%
Mixed	Import stored data	20.0%
Position	Tasks – ^3 (Cubed)	% Errors
Mixed	Choose a song from a playlist	11.2%
Mixed	Add a new song in/to a playlist	16.7%
Vertical	Change the application's appearance	10.6%
Horizontal	Choose a new theme	14.0%
Position	Tasks – Shuffle	% Errors
Mixed	Add a new task	12.9%
Mixed	Delete all activities	14.3%
Vertical	Backup data	14.3%
Mixed	Open the help option	20.0%

5.4. Smartphones' physical characteristics analysis

The following analysis verifies the existence of contextual variables interference related to smartphone characteristics, such as, screen resolution and size. In order to carry out this evaluation, the tasks executions were investigated considering the smartphones' characteristics. The data presented in Figure 14 allows identifying that the screen resolution influences significantly in the tasks execution speed, that is, the higher the resolution, the faster the tasks are concluded. We can observe, as for the application Cubed, which the high resolution increases, in average 26.03%, the task speed when compared to the low resolution. In the Mileage application, this difference is 19.66% and, in the Shuffle, 17.17%.
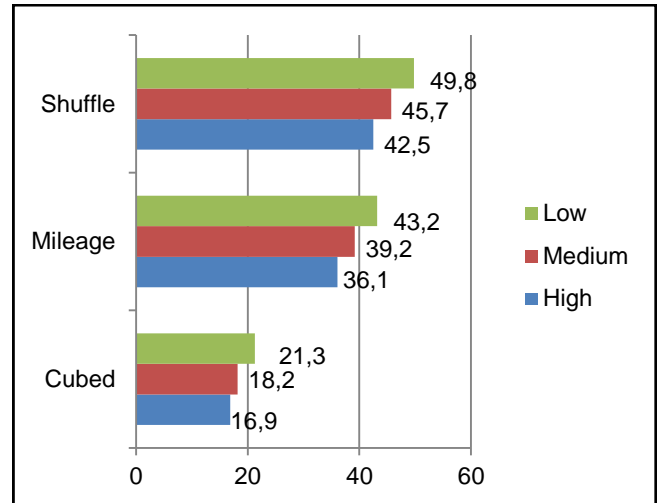

Figure 14. Tasks Execution Speed due to screen resolution (in seconds)

The same analysis made earlier was designed to verify the screen size influence on the users' interactions. When observing Figure 15, it is possible to see that the screen size is another contextual variable which influences the performance of users. In the Cubed application, the average speed for the task execution decreases around 4.1 seconds when compared to the use of small screen smartphones. In the Mileage application, this difference is apparent around 6.9 seconds, and, in the Shuffle, the difference is 10.9 seconds.
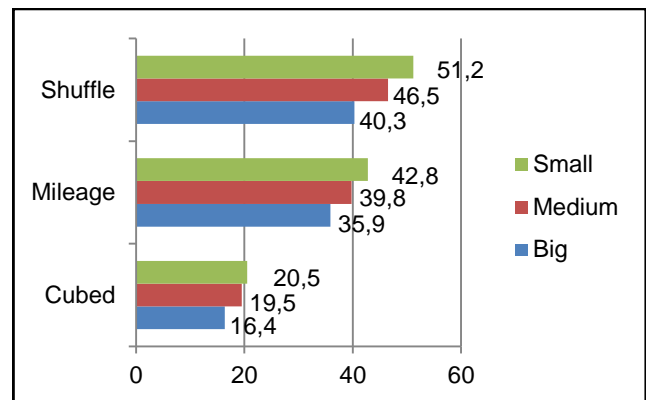

Figure 15. Tasks execution speed due to the screen size (in seconds)

A fact observed in the smartphone market is that, normally, the phones with smaller screens also have lower resolution. Thus, the users' performance was observed considering the two variables simultaneously. The metrics used to measure the performance was the percentage of tasks completed with error. The graph in Figure 16 shows that the smaller the size and the lowest the resolution of the

smartphones screen, more errors are found in the executed tasks. The difference between the extremes, that is, big screens with high resolution compared to small screens with low resolution, is 9.3% of tasks executed with errors.
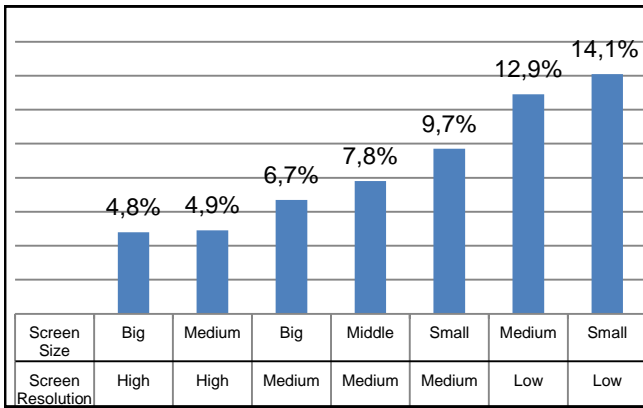
Figure 16. Relationship between the screen size and resolution and the errors percentage

5.5. Profile of participants analysis

When analysing the rate of tasks executed with errors along with the profile of participants, an intriguing fact was observed. The occurrence of errors in the low social class is greater than in the other classes. In order to search for an explanation for this result, the kind of device used in the experiment by these participants was investigated. The conclusion was that the rate of errors was not related to the users' purchasing power, but to the low resolution of the device's screen. As the majority of the people with low purchasing power used low resolution smartphones, an isolated analysis of the social class can lead to wrong conclusions.
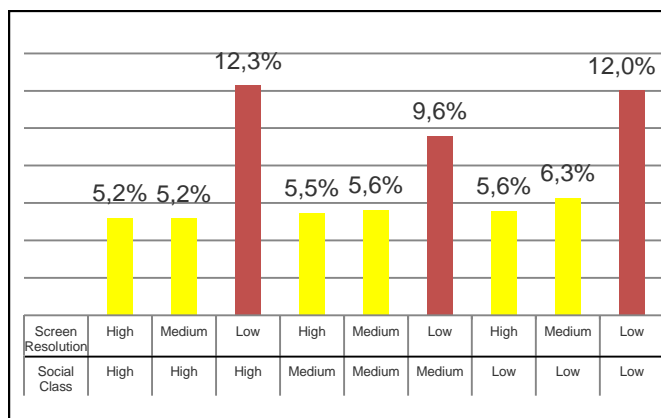
Figure 17. Relationship between the error rates due to purchasing power X screen resolution

In Figure 17, it is observed that, regardless of the purchasing power, the errors are more frequent when low resolution smartphones are used. This analysis characterizes one of the potentialities of the UXEProject infrastructure as it permits to associate different contextual factors in a single evaluation, decreasing thereby the possibility of wrong conclusions.

5. Conclusions and Prospects

Based on the data presented in Section 2 of this article, the conclusion is that the majority of experiments made to evaluate the usability of applications for smartphones use surveys to collect the data and there is no correlation between the contextual variables and the usability problems observed. This fact is contrary to the expectations of many researchers in this area.

The results obtained in the experiment showed that the UXEProject infrastructure is a good solution for the investigation of usability problems associated to different types of data, highlighting the data collection using the smartphones' sensors.

With the experiments' results, it is observed that approximately 70% of the interactions occur when users are stationary, having the device in a single position and with a normal environment luminosity. However, when these contextual factors change, the users make more mistakes and take longer to execute the tasks. This information suggests that the applications should, for example (i) make interactions impractical in positions which offer more probability of errors, forcing users to interact in an appropriate position; (ii) detect the external luminosity and try to balance the luminosity radiated by the device in order to guarantee a good visualization; and (iii) identify the user's movement and only enable the most usual functionalities, decreasing the visual pollution.

Another important observation concerns the smartphones' setup interference in the users' performance. Furthermore, it was proved that the correlation of different kinds of information are important for the conclusion of the results, as seen in the relationship between the errors rate and low purchasing power people.

As prospects for the future, it is intended to incorporate other sensors to the UXEProject infrastructure, aiming to conduct new investigations involving different contextual factors.

References

[1] Weiser, M. (1991) The computer for the 21st century. Scientific American, 265(3): 66-75.

[2] Vieira, V., Tedesco, P., Salgado, A. C. (2011) Designing context-sensitive systems: An integrated approach. In Expert Systems with Applications, 38(2): 1119-1138.

[3] Hansen, J. (2012). An Investigation of Smartphone Applications: Exploring usability aspects related to Wireless Personal Area Networks, Context-awareness, and Remote Information Access. Doctor' thesis, School of Information Systems, Computing and Mathematics – Brunel University.

[4] Balagatas-Fernandez, F. and Hussmann, H. (2009) Evaluation of User-Interfaces for Mobile Application Development Environments. In Human-Computer Interaction. New Trends. New Trends, v. 5610, Springer: 204-213.

[5] Jensen, K. L. and Larsen, L. B. (2007) Evaluating the Usefulness of Mobile Services Based on Captured Usage Data from Longitudinal Field Trials. In proceedings of 4th International Conference on Mobile Technology, Application and Systems. ACM: 675-682.

[6] Zhang, D. and Adipat, B. (2005) Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications. International Journal of HCI, 18(3): 293-308.

[7] Mallick, M. (2003). Mobile and wireless design essentials. 2nd ed. (New York: Wiley).

[8] Kawalek, J., Stark, A., Riebeck, M. (2008) A New Approach to Analyze Human-Mobile Computer Interaction. Journal of Usability Studies, 3(2): 90-98.

[9] Coursaris, C. K. and Kim, D. J. (2011) A Meta-Analytical Review of Empirical Mobile Usability Studies. Journal Usability Studies, 6(3): 117-171.

[10] Burigat, S., Chittaro, L. and Gabrielli, S. (2008) Navigation techniques for small-screen devices: An evaluation on maps and web pages. Int. Journal of HC Studies, 66(2): 78-97.

[11] Sodnik, J., Dicke, C., Tomazic, S. and Billinghurst, M. (2008) A user study of auditory versus visual interfaces for use while driving. Int. Journal of HC Studies, 66(5): 318-332.

[12] Fitchett, S. and Cockburn, A. (2009) Evaluating reading and analysis tasks on mobile devices: A case study of tilt and flick scrolling. In The 21st Annual Conference of the Australian CHI, 255-232.

[13] Chin, A. and Salomaa, J. P. (2009) A user study of mobile web services and applications from the 2008 Beijing Olympics. In 20th ACM conference on Hypertext and Hypermedia: 343-344.

[14] Lai, J., Vanno, L., Link, M., Pearson, J., Makowska, H., Benezra, K. and Green, M. (2009) Life360: Usability of Mobile Devices for Time Use Surveys. In. American Association for Public Opinion Research Annual Conference, Hollywood, FL: 5582–5589.

[15] Ebner, M., Stickel, C., Scerbakov, N. and Holzinger, A. (2009) A Study on the Compatibility of Ubiquitous Learning (u-Learning) Systems at University Level. Universal Access in HCI. Applications and Services. Springer Lecture Notes in Computer Science, v. 5616: 34-43.

[16] Hansen, J. and Ghinea, G. (2009) Multi-platform Bluetooth remote control. IEEE International Conference for Internet Technology and Secured Transactions: 1-2.

[17] BØdker, M., Gimpel, G. and Hedman, J. (2009) Smart Phones and their Substitutes: Task-Medium Fit and Business Models. In 8th Int. Conf. on Mobile Business, IEEE Computer Society: 24-29.

[18] Kim, S., Lee, I., Lee, K., Jung, S., Park, J., Kim, Y. B., Kim, S. R. and Kim, J. (2010) Mobile web 2.0 with multi-display buttons. Communications of the ACM, 53(1): 136-141.

[19] Li, Y. M., and Yeh, Y. S. (2010) Increasing trust in mobile commerce through design aesthetics. Computers in Human Behavior, 26(4): 673-684.

[20] Maly, I., Mikovec, Z. and Vystrcil, J. (2010) Interactive analytical tool for usability analysis of mobile indoor navigation application. In: The 3rd International Conference on Human System Interaction, IEEE: 259-266.

[21] GrØnli, T., Hansen, J. and Ghinea, G. (2010) Android, Java ME and Windows Mobile interplay: The Case of a Context-Aware Meeting Room. 24th International Conference on Advanced Information Networking and Applications Workshops (WAINA), IEEE Computer Society: 920-925.

[22] Kang, Y. M., Cho, C. and Lee, S. (2011) Analysis of factors affecting the adoption of smartphones. In Technology Management Conference (ITMC), IEEE International: 919-925.

[23] Fetaji. B, Ebibi, M. and Fetaji, M. (2011) Assessing Effectiveness in Mobile Learning by Devising MLUAT Methodology. International Journal of Computers and Communications, 5(3): 178-187.

[24] Hegarty, R. and Wusteman, J. (2011) Evaluating EBSCOhost Mobile. Library Hi Tech, 29 (2): 320-333.

[25] GrØnli, T., Hansen, J. and Ghinea, G. (2011) Integrated Context-aware and Cloud-based Adaptive Application Home Screens for Android Phones. 14th International Conference on HCI. Springer: 427-435.

[26] Sparkes, J., Valaitis, R. and Mckibbon, A. (2012) A Usability Study of Patients Setting Up a Cardiac Event Loop Recorder and BlackBerry Gateway for Remote Monitoring at Home. Official Journal of the American Telemedicine Association, 18(6): 484-490.

[27] Bradley, P. R., et al. (2012) Smartphone Application for Transmission of ECG Images in Pre-Hospital STEMI Treatment. Proceedings of the Systems and Information Design Symposium. IEEE: 118-123.

[28] Schaub, F., Deyhle, R. and Weber, M. (2012) Password Entry Usability and Shoulder Surfing Susceptibility on Different Smartphone Platforms. Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia, ACM.

[29] Kirwan, M., Duncan, M. J., Vandelanotte, C., Mummery, W. K. (2012) Using smartphone technology to monitor physical activity in the 10,000 Steps program: a matched case-control trial. Journal of Medical Internet Research. Available in: http://www.jmir.org/2012/2/e55/.

[30] Spyridonis, F. GrØnli, T., Hansen, J. and Ghinea, G. (2012) Evaluating the Usability of a Virtual Reality-Based Android Application in Managing the Pain Experience of Wheelchair Users. Proceedings of 34th Annual International Conference of the IEEE EMBS - Engineering in Medicine and Biology Society: 2460-2463.

[31] Silva, P.A., Nunes, F., Vasconcelos, A., Kerwin, M. Montinho, R., Teixeira, P. (2013) Using the Smartphone Accelerometer to Monitor Fall Risk while Playing a

Game: The Design and Usability Evaluation of Dance! Don't Fall. D.D. Schmorrow and C.M. Fidopiastis (Eds.): AC/HCII 2013, LNAI 8027, Springer-Verlag Berlin Heidelberg, 754-763.

[32] Spook, J. E., Paulussen, T, Kok, G. and Empelen, P. V. (2013) Monitoring Dietary Intake and Physical Activity Electronically: Feasibility, Usability, and Ecological Validity of a Mobile-Based Ecological Momentary Assessment Tool. Journal of Medical Internet Research, Published online. DOI: 10.2196/jmir.2617. Available at: <http://www.jmir.org/2013/9/e214/> (accessed 5 May 2014).

[33] Ben-Zeev, D., Kaiser, S. M., Brenner, C. J., Begale, M., Duffecy, J., Mohr, D. C. (2013) Development and usability testing of FOCUS: A smartphone system for self-management of schizophrenia. Psychiatric Rehabilitation Journal, 36, 289-296.

[34] Othmana, M.K., Petrie, H. and Power, C. (2013) Measuring the usability of a smartphone delivered museum guide. Elsevier: Proc. - Social and Behavioral Sciences, 97, 629-637.

[35] Okazaki, S., Mendez, F. (2013) Exploring convenience in mobile commerce: Moderating effects of gender. Elsevier: Computers in Human Behavior, 29, 1234-1242.

[36] Ferreira, F., Almeida, N., Rosa, A. F., Oliveira, A., Teixeira, A., Pereira, J. C. (2014) Elderly centered design for Interaction – the case of the S4S Medication Assistant. Elsevier: Procedia Computer Science, 27, 398-408.

[37] Reynoldson, C., Stones, C., Allsop, M., Gardner, P., Bennett, M., Closs, J.S., Jones, R., Knapp, P. (2014) Assessing the Quality and Usability of Smartphone Apps for Pain Self-Management. Wiley Periodicals, Pain Medicine, Available at: <http://onlinelibrary. wiley.com/ doi/10.1111/pme. 12327/pdf > (accessed 5 May 2014).

[38] Barros, A. C., Leitão, R., Ribeiro, J. (2014) Design and evaluation of a mobile user interface for older adults: navigation, interaction and visual design recommendations. Elsevier: Procedia Computer Science, 27, 369-378.

[39] Campbell, B. A., Tossell, C. C., Byrne, M. D., Kortum, P. (2014). Toward More Usable Electronic Voting: Testing the Usability of a Smartphone Voting System. Human Factors: The Journal of the Human Factors and Ergonomics Society published online. DOI: 10.1177/0018720813519266.

[40] Molina, A. I, Redondo, M. A., Lacave, C, Ortega, M. (2014) Assessing the effectiveness of new devices for accessing learning materials: An empirical analysis based on eye tracking and learner subjective perception. Computers in Human Behavior, 31, 475-490.

[41] Courage, C. and Baxter, K. (2005) Understanding your user: a practical guide to user requirements, methods, tools, and techniques. 2nd ed. (Sao Francisco: Morgan Kaufmann Publishers).

[42] Kronbauer, A. H., Santos, C. A. S. and Vieira, V. (2012) Smartphone Applications Usability Evaluation: A Hybrid Model and Its Implementation. Proceedings of 4th International Conference on Human-Centred Software Engineering, Springer: 146-163.

[43] Ickin, S., Wac, K., Fiedler, M., Janowski, L., Hong, J. and Dey, A. K. (2012) Factors Influencing Quality of Experience of Commonly Used Mobile Applications. IEEE Communications Magazine, 5(4): 48-56.

[44] Kronbauer, A. H, Santos, C. A. S. (2014) Impact of Contextual Factors on Smartphone Applications Use. Mobile and Ubiquitous Systems: Computing, Networking, and Services Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering Vol. 131, 667-679.

[45] Sharp, H., Rogers, Y., Preece, J. (2007) Interaction design: beyond human-computer interaction. 2nd ed. (New York: John Wiley & Sons).