



Gonçalo Lopes Ferreira

Licenciado em Ciências da Engenharia e Gestão Industrial

User-Centered Design of the Interface Prototype of a Business Intelligence Mobile Application

Dissertação para obtenção do Grau de Mestre em
Engenharia e Gestão Industrial

Orientadora: Professora Doutora Isabel Lopes Nunes,
Professora Auxiliar, Faculdade de Ciências e Tecnologia
da Universidade Nova de Lisboa

Júri:

Presidente: Prof. Doutor Virgílio António Cruz Machado

Arguente: Prof. Doutor Nuno Manuel Robalo Correia

Vogais: Prof. Doutora Isabel Maria Nascimento Lopes Nunes

Eng^o Bruno Miguéns Mourão



Junho 2013

**Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia**

**User-Centered Design of the Interface Prototype of a Business Intelligence Mobile
Application**

Gonçalo Lopes Ferreira

Licenciado em Ciências da Engenharia e Gestão Industrial

Dissertação para obtenção do Grau de Mestre em Engenharia e Gestão Industrial

Orientadora: Professora Doutora Isabel Lopes Nunes,
Professora Auxiliar, Faculdade de Ciências e Tecnologia da
Universidade Nova de Lisboa

Março 2013

User-centered design of the interface prototype of a business intelligence mobile application

Copyright © Gonçalo Lopes Ferreira, FCT/UNL e UNL

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limitações geográficas, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Acknowledgments

To my dissertation advisor, Professor Doctor Isabel Lopes Nunes, for the availability, encouragement, support, text reviews and for always pointing me in the right direction.

To my internship advisors, Bruno Mourão and João Ferreira for all the help, knowledge, advice, commitment and availability.

To everyone at the office: Sofia, José, Daniela, Rita, Inês Torres, Inês Dias, Mariana, Hugo, Vanessa and Pedro for the help and for making me feel like a part of the team.

To the users and participants of the research process: Miguel Andrade, Luís Tomás, Vítor Santos, Paulo Nazaré and José Grilo for their availability and interest in helping this dissertation reach its goals.

To Alfredo Rodrigues, for the interest in making this dissertation possible.

To Professor Doctor Rui Rodrigues, for his wise words and for believing in me.

To Joana, for the support, faith, kindness and joy.

To my college friends, The AVC: Coelho, Diogo, DJ, Gus, Johnny, Nakata, Rato, Salgado and Zé for the moral support and all the good times spent together in these last few years.

To my family for all the love, kindness and support.

To my parents, for all their support, dedication, concern and faith and for educating me and providing me with all the comfort needed to focus on my academic course. I owe you everything, Thank you.

*“Everybody is a genius.
But if you judge a fish by its ability to climb a tree,
it will live its whole life believing that it is stupid”*

Albert Einstein

Abstract

In a society in constant technological evolution, companies try to equip themselves with tools that allow them to achieve, or maintain, the leadership position in the markets in which they compete in. Success often lies in the ability to exploit the existent Business Intelligence (BI) in the best possible way. Moreover, mobile devices such as smartphones and tablets assume themselves as elements with an increasingly preponderant use in the everyday life. In the convergence of these two factors, emerges the need to develop mobile “management facilitators”, capable of providing companies’ workers the access to BI, anywhere and at a few touches distance. To make this possible, not only does the correct information needs to be selected, but also organized and presented in a highly intuitive and easy to use way.

It’s in this context that the work present in this dissertation emerges: User-Centered Design of the interface prototype of a Business Intelligence mobile application for a company in the retail industry. Thus, the goal of this dissertation resides in the development of an adequate interface, through the continuous interaction with a group of representative users in activities such as users’ needs assessment, interface design, heuristic evaluation and usability tests.

From meetings with the users, 29 needs were identified for the BI application. These needs were later converted into functional requirements which originated two prototypes, one for smartphones and the other for tablets. These were subjected to a heuristic evaluation and tested through the application of the Cognitive Walkthrough method to representative users, to collect performance and satisfaction metrics.

It was concluded that the designed interfaces were in accordance with 14 of the 16 heuristics, which led to three modifications on the interfaces. The Cognitive Walkthrough results showed that the interfaces are intuitive since all the tasks were completed with 100% success, in reasonable times and with a number of actions close to the ideal.

Keywords: User-Centered Design; Cognitive Walkthrough; Heuristic Evaluation; Prototype; Application for mobile equipment; Business Intelligence.

Resumo

Numa sociedade em constante evolução tecnológica, as empresas tentam munir-se de ferramentas que lhes permitam atingir, ou manter, a liderança nos mercados nos quais competem. O sucesso muitas vezes reside na capacidade de explorar da melhor forma a informação de gestão existente. Por outro lado, equipamentos móveis como *smartphones* e *tablets* assumem-se como elementos de uso cada vez mais preponderante no quotidiano. Na convergência destes dois fatores, surge a necessidade de desenvolver “facilitadores de gestão” móveis, que disponibilizem aos trabalhadores das empresas o acesso a *Business Intelligence* em qualquer lado e à distância de poucos toques. Para tal, não só é necessário selecionar a informação correta mas também organizá-la e apresentá-la de uma forma altamente intuitiva e de fácil utilização.

É neste contexto que surge o trabalho desenvolvido nesta dissertação: *User-Centered Design* de um protótipo da interface de uma aplicação móvel de *Business Intelligence* para uma empresa da indústria do retalho. Assim, esta dissertação tem como objetivo o desenvolvimento de uma interface adequada, através da constante interação com um grupo de utilizadores representativos do público-alvo em atividades como o levantamento de necessidades, desenho de interfaces, avaliação heurística e testes de usabilidade.

Em reuniões com os utilizadores foram identificadas 29 necessidades para a aplicação móvel de BI. As necessidades foram posteriormente convertidas em requisitos funcionais que por sua vez originaram dois protótipos, um para *smartphone* e outro para *tablet*. Estes foram submetidos a uma avaliação heurística e testados através da aplicação do método *Cognitive Walkthrough* a utilizadores representativos de modo a recolher métricas de *performance* e satisfação.

Concluiu-se que as interfaces desenhadas estavam em conformidade com 14 das 16 heurísticas, o que levou a três modificações nas interfaces. Os resultados do *Cognitive Walkthrough* mostraram que as interfaces são intuitivas visto que todas as tarefas foram concluídas com 100% de sucesso, em tempos razoáveis e com um número de ações próximo do ideal.

Palavras-Chave: *User-Centered Design*; *Cognitive Walkthrough*; Avaliação Heurística; Protótipo; Aplicação para equipamentos móveis; *Business Intelligence*.

Table of Contents

1.	Introduction.....	1
1.1.	Context	1
1.2.	Objectives	2
1.3.	Methodology	3
1.4.	Structure of the dissertation	4
2.	Literature review	7
2.1	Business intelligence	7
2.2	Mobile business intelligence	8
2.3	User-centered design	8
2.4	Usability testing	12
3.	Interface prototypes development	19
3.1.	User’s needs assessment	19
3.2.	System functional requirements definition	22
3.3.	Preparations for the interface design	30
3.4.	Interface design	37
4.	Usability evaluation and testing	53
4.1.	Interface simulator creation	53
4.2.	Heuristic evaluation	54
4.3.	Cognitive walkthrough	65
4.4.	Analysis and discussion of results	74
5.	Conclusions and suggestions for future work	87
5.1.	Conclusions	87
5.2.	Future work	89
	References	91
	Appendices	97
	Appendix A – Overview of the process: From the user’s needs to the interface design .	99
	Appendix B - Gantt chart of the User-Centered Design activities	103
	Appendix C - Questionnaire handed to the Cognitive Walkthrough participants	107
	Appendix D - Cognitive Walkthrough testing results by participant	111

List of Figures

Figure 1.1 – BI application development methodology overview.	3
Figure 2.1 – The activities of the UCD according to ISO 13407 (1999).....	9
Figure 3.1 – Seven users raised 29 needs.	20
Figure 3.2 – User’s needs prioritization criteria.	23
Figure 3.3 – Final prioritization of the user’s needs.	23
Figure 3.4 – Outcome of the final meeting.....	29
Figure 3.5 – The two approaches of prototyping (adapted from (Nielsen, 1993)).	32
Figure 3.6 – Comparison between iOS segmented control and Android drop-down menu.	34
Figure 3.7 – Similarities between Facebook on iOS and Android.	35
Figure 3.8 – Tablet preferred orientation in finance and business applications advertising (adapted from (Park, 2011)).	36
Figure 3.9 – “Login” screen comparison between the smartphone and tablet interfaces.	38
Figure 3.10 – “Home” screen comparison between the smartphone and tablet interfaces.....	38
Figure 3.11 – Navigation between the "Home" and the "KPI 1" screens in the smartphone interface.	40
Figure 3.12 – “KPI 1a” screen comparison between the smartphone and tablet interfaces.	40
Figure 3.13 – Several sidebar design versions.	42
Figure 3.14 – Expanding the KPI sidebar.....	43
Figure 3.15 – Connection between the “KPI 1a” and the “KPI 1a Chart” screens in the smartphone interface.	43
Figure 3.16 – Navigation in the “KPI 1a Chart” screen.	44
Figure 3.17 – “Notifications Menu” screen comparison between the smartphone and tablet interfaces.	45
Figure 3.18 – "Clear" button functionality.	46
Figure 3.19 – Notifications “Configuration” screen comparison between the smartphone and tablet interfaces.....	47
Figure 3.20 – Spinner popup to select the desired option.	48
Figure 3.21 – Saving notifications configurations.....	48
Figure 3.22 – “Product File” screen comparison between the smartphone and tablet interfaces...	49
Figure 3.23 – Both ways to access the product file.	49
Figure 3.24 – “Shortcuts” screen comparison between the smartphone and tablet interfaces.	50
Figure 3.25 – Logout confirmation.	51
Figure 3.26 – Six user's needs originated the design of 199 screens.....	51
Figure 4.1 – Interface simulation using PowerPoint.	53
Figure 4.2 – "Configurations Saved" notification	56
Figure 4.3 – Animation indicating that the refreshing is in process.	56
Figure 4.4 – Example of an attempt to reduce users' memory load.....	58
Figure 4.5 – “Product Not Found” error message.	60
Figure 4.6 – Three different sizes for touchable targets (adapted from (Microsoft, 2012)).	62
Figure 4.7 – Target and space sizes in a critical area of the smartphone interface.....	62
Figure 4.8 – Icons used in the interfaces.	63
Figure 4.9 – New version for the tab icons.....	64
Figure 4.10 – "Magnifying Glass" icon modification.	64
Figure 4.11 – "Mr. Tappy" (a) and “MOD 1000” (b) kits to film handheld devices (Bowman, 2012) and (Sauro, 2012).	73

Figure 4.12 – A participant performing the Cognitive Walkthrough on the smartphone interface prototype simulation.	74
Figure 4.13 – Task time results for the tablet (a) and the smartphone (b) prototypes.	76
Figure 4.14 – Errors results (number of errors) for the tablet (a) and the smartphone (b) prototypes.....	78
Figure 4.15 – Errors results (participants who made errors) for the tablet (a) and the smartphone (b) prototypes.....	78
Figure 4.16 – Reason for the high number of "Type B" errors on task 6.	79
Figure 4.17 – Tab bar of the smartphone interface.	79
Figure 4.18 – Initial version of the method to access charts.....	80
Figure 4.19 – Second version of the method to access charts.....	81
Figure 4.20 – Efficiency results for the tablet (a) and the smartphone (b) prototypes.....	81
Figure 4.21 – ASQ results for the tablet (a) and the smartphone (b) prototypes.	83
Figure 4.22 – SUM results for the tablet and the smartphone prototypes.....	84
Figure C.1 – Page 1 (Personal data) of the questionnaire handed to the Cognitive Walkthrough participants.....	109
Figure C.2 – Pages 2 to 7 (ASQ) of the questionnaire handed to the Cognitive Walkthrough participants.....	109
Figure C.3 – Page 8 (SUS) of the questionnaire handed to the Cognitive Walkthrough participants.....	110

List of Tables

Table 3.1 – Fraction of the organized list of collected user’s needs.....	21
Table 3.2 – List of currently used BI applications and the KPIs accessible from them.	22
Table 3.3 – Current constraints vs. future benefits analysis for user’s need 1.	25
Table 3.4 – Current constraints vs. future benefits analysis for user’s need 11.	25
Table 3.5 – Current constraints vs. future benefits analysis for user’s need 19.	26
Table 3.6 – Converting user’s need 1 into system functional requirements.....	27
Table 4.1 – List of heuristics and guidelines used in the evaluation of the interface.	55
Table 4.2 – Common usability study scenarios and the appropriate metrics to collect in each case (adapted from (Tullis & Albert, 2008)).	68
Table 4.3 – Testing results example (Participant 7).	75
Table 4.4 – Task time results for the tablet and the smartphone prototypes.	76
Table 4.5 – SUS results for the tablet and the smartphone prototypes.....	85
Table D.1 – Testing results of participant 1.	113
Table D.2 – Testing results of participant 2.	113
Table D.3 – Testing results of participant 3.	114
Table D.4 – Testing results of participant 4.	114
Table D.5 – Testing results of participant 5.	115
Table D.6 – Testing results of participant 6.	115
Table D.7 – Testing results of participant 7.	116
Table D.8 – Testing results of participant 8.	116

List of Abbreviations

ASQ	After-Scenario Questionnaire
BI	Business Intelligence
EDA	Enterprise Digital Assistant
GUI	Graphical User Interface
KPI	Key Performance Indicator
OS	Operating System
SUM	Single Usability Metric
SUS	System Usability Scale
UCD	User-Centered Design
UX	User Experience

1. Introduction

1.1. Context

Nowadays, mobile devices such as smartphones and tablets have an increasingly important role in society. All around the world, people are trading in their old mobile phones in favor of brand new smartphones. According to a recent report, the number of smartphones being used worldwide exceeded the 1 billion mark for the first time ever (Five Star Equities, 2012). This pattern can be observed in Portugal as well. In the first quarter of 2012, 307 thousand smartphones have been sold, which means a sales growth of 40% compared with the same period of 2011 (Fernandes, 2012). Tablets are also finding their place in people's lives. Recent studies reveal that worldwide tablet sales are predicted to rise to 165.9 million units in 2013 (IDC, 2012). Once again, Portugal follows this trend. Total tablet sales in Portugal were estimated to reach 220 thousand units at the end of 2012 and 320 thousand in 2013. However, despite this growth, the volume sold in the Portuguese market is much smaller when compared to other countries with a similar amount of inhabitants (Dinheiro Vivo, 2012). The numbers speak for themselves and the natural trend is for the number of people who own smartphones and tablets to keep rising.

Aside from personal use, these devices are ideal for increasing the productivity of multi-tasking business professionals (Kavan, 2011). With this in mind, it is not a surprise that companies often adapt the way they work, to take advantage of these technologies hoping the transition would increase their revenues. Smartphones have been playing a key role in industries such as the military, logistics, manufacturing, distribution, retail, and many others. For instance, in the retail industry, smartphones and tablets are being used by companies in activities such as barcode changing, order processing, stock location, supply chain management and communication improvement (Roche, 2012).

All of these tasks are easier to accomplish due to the incorporation of smartphones and tablets in the companies way of working, which results in an increase in employee productivity while also helping them achieve higher efficiency levels (Roche, 2012).

However, turning processes and activities simpler and easier doesn't happen merely by introducing these devices in the daily business. Instead, it's due mainly to the usability of these devices, designed with the aim of improving the users' experience and making their tasks easier to accomplish. Also, for applications developed specifically for company internal use, usability is a matter of employee productivity. If the users lose themselves in applications with low usability, it translates into lost time, which in turn translates into money wasted by the employer (Nielsen, 2012).

Introduction

Having usability in consideration in every stage of product development is essential to obtain an easy to use product, capable of improving the user's productivity. However, when it comes to multitouch interfaces such as smartphones and tablets, there isn't much body of knowledge approaching this matter, perhaps due to the young nature of the devices. Despite this situation, there is a vast literature regarding guidelines and good practices for usability, which, coupled with the guidelines provided by the creators of smartphones and tablets operating systems (OS) can somewhat ensure usability for touch and multitouch systems (Simões-Marques & Nunes, 2012).

It's in this context that the opportunity presented in this dissertation appeared. A retail company decided that the prospect of providing their store managers a solution that would grant them mobility while working was very appealing and seemed like an achievable project. The referred profession requires people to be in constant motion, moving between their office and other parts of their working place. However, they are currently forced to be at a specific place such as their office in order to access data necessary for them to make decisions, thus the importance of developing a mobile Business Intelligence (BI) application.

The choice for materializing this concept fell on touchscreen systems, because they're more suited for information retrieval rather than data entry (Meredith, 2008) and also because of their intuitive, easy to use and flexible nature (Simões-Marques & Nunes, 2012) ideal for the type of user role in question.

1.2. Objectives

The main goal of this dissertation is the development, evaluation and testing of two interface prototypes, one for smartphones and the other for tablets, of a mobile touchscreen BI application for store managers at a retail company. The interface prototypes intend to provide developers with a good base upon which they can create the interface of the real BI application. The application should be capable of granting the company's store managers mobility, while giving them access to management information to assist them in making decisions and also the means to detect problems and inefficiencies in the production process. By doing such, it is expected an increase in the users' productivity as well as improvements in their working method.

Both interface prototypes should be developed according to the user's needs, take advantage of the standard mobile devices capabilities, such as animations, transitions and navigation; and ultimately become the foundation of the real BI application.

Finally, usability testing will take place under a method called Cognitive Walkthrough where participants are invited to complete a number of tasks that aim to represent real situations they would encounter while using the BI application in development on a workday.

1.3. Methodology

The methodology to develop the BI application followed a User-Centered Design (UCD) approach and consisted of four main phases: user's needs assessment, system functional requirements definition, interface prototypes development and usability evaluation and testing. Each phase encompassed various activities. Their length ranged from only a few hours to more than a month. The four phases, as well as the main activities contained in them, are depicted in Figure 1.1.

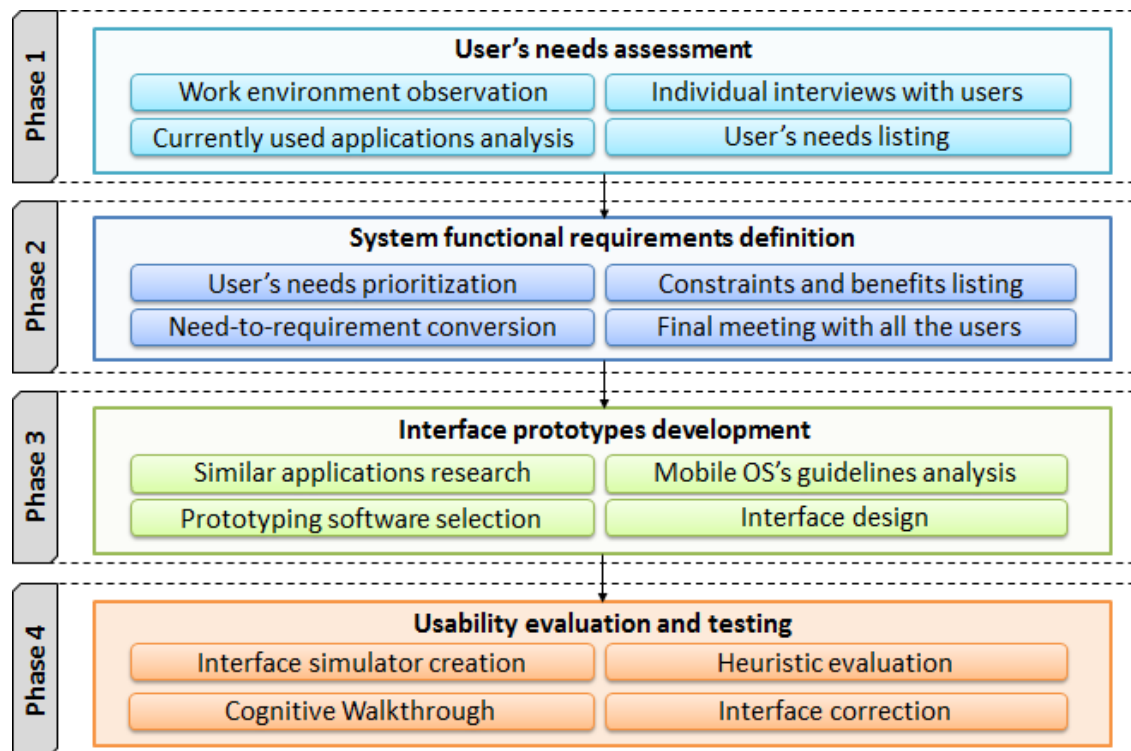


Figure 1.1 – BI application development methodology overview.

Phase 1, as the name suggests, involved knowing the target user as well as its needs (through individual interviews and work environment observation). At this point the users were also asked about:

- What is the most important information subject of a daily analysis;
- Which applications are accessed in order to retrieve the most important information.

This phase ended with the listing of all the needs collected from the interviews.

In the second phase, the needs listed in phase 1 were prioritized and “converted” into system functional requirements. Current constraints and future benefits for each user's need were also listed. After defining all the requirements, a final meeting with all the users took place. This meeting had the main purpose of promoting the discussion about the needs prioritization. Also, it

served to discuss and reach an agreement on the information and functionalities to be included on the application.

In phase 3, the design of both the smartphone and the tablet interfaces took place. Before actually designing the interfaces, a market research of similar applications was done to gain inspiration and also some knowledge on the matter. These interfaces designs were based on usability principles as well as guidelines provided by the OS' developers such as Android™ and Apple® that provide step-by-step tutorials on how to develop good applications for their systems.

The fourth and final phase refers to usability evaluation and testing. Firstly, a simulation of the interfaces was created. Then both prototypes were evaluated according to usability heuristics and a group of participants were asked to take part in a Cognitive Walkthrough, where they were presented a number of tasks and tried to complete them on the interface prototypes. Finally, the results of the tests were processed and analyzed with the purpose of improving the interfaces design. Being a product intended to serve professional needs, it's extremely important to have a good usability and thus, to be easy to use.

The overview of the all the steps in the process between collecting the user's needs and designing the interfaces, as well as the Gantt chart representing all the UCD activities, are both presented in the appendices section (appendices A and B, respectively).

1.4. Structure of the dissertation

This dissertation is organized in five chapters which will be briefly described next.

The first chapter, of introductory nature, contextualizes the reader in the expanding world of smartphones and tablets and their growing usage in the industrial environment, lists the objectives of the dissertation and explains in a very concise way the methodology used to achieve the objectives.

The second chapter presents the literature review. The main concepts present in the dissertation such as Business Intelligence and User-Centered Design are explained and their importance is illustrated with several examples found on the literature.

The third and fourth chapters describe all the UCD activities that took place in order to accomplish this dissertation's objectives. Both chapters are structured in a way that reflects the order in which the activities were executed.

The third chapter refers to the first three phases of the BI application development (user's needs assessment, functional requirements definition, currently used applications analysis and interfaces design). It describes how the user's needs were collected, how many needs there were and how

Introduction

they were “converted” into functional requirements. It also provides an analysis of the currently used applications. Furthermore, many screens representing the different sections of the application are displayed along with detailed explanations of the factors that contributed to their design.

The fourth chapter approaches the last phase of development which comprises the heuristic evaluation and the usability testing of the prototypes. Concerning the first, it describes the criteria behind selecting appropriate heuristics to evaluate the prototypes and shows the results from such evaluation. Regarding the second, it lists the metrics collected from the tests and describes how the tests took place. Finally the results from the tests are analyzed and some modifications done to the interfaces in accordance with the results.

The final chapter concludes this dissertation by briefly recalling what was done to achieve the objectives, what conclusions were drawn and also approaches what could be the future work for similar cases.

2. Literature review

In the ever-changing retail industry, companies need to be constantly adapting themselves to consumer demand and use all the tools at their disposal to increase their competitiveness and the efficiency of their operations in order to boost their revenue. Retailers have a massive amount of information on their hands and sometimes it becomes hard to keep track of what is important and valuable (MicroStrategy, 2007). One common challenge that retailers face is the availability of the right information, at the right place and at the right time (Moinuddin, 2006). The line between success and failure sometimes lies in finding the correct way to harness such information, and use it to the company's advantage.

2.1 Business intelligence

Given the constant need for computer systems that support the analysis of large amounts of data from various perspectives, containing current and historical data, as well as the integration of the data from several operational systems, it emerged in the 80s the concept of Business Intelligence (Gonçalves, Santos, & Cruz, 2011). As a term, BI replaced decision support, executive information systems, and management information systems (Thomsen, 2003). BI systems provide actionable information delivered at the right time, at the right location and in the right form to assist decision makers in making accurate business decisions (Nagash, 2004).

To this extent, it's natural that retail companies set their sights on solutions that provide them the ability to easily access the most important and critical information in a clean and organized way. The implementation of BI applications can be found in many industries. The following paragraphs describe some examples of the implementation of BI solutions across a variety of industries.

For example, the Credit Information Bureau, a department of the State Bank of Pakistan implemented BI applications in order to be able to provide fast, dynamic and accurate analysis on individual or group borrowers (Nadeem & Jaffri, 2004).

A different study, proposes the implementation of a BI system in factories' shop floor in order to organize the large quantities of data it generates. This study aims to aid managers in decision-making, thereby enhancing the company's competitiveness. To that extent, a prototype (with simulated data) was developed to test the viability of that proposal and it was concluded that the flexibility of the system would provide managers with a good support for decision making in search for better results (Fortulan & Filho, 2005).

Another study, which researched the importance of having BI in small and medium-sized enterprises and attempted to implement BI applications in that context, concluded that BI would

allow small and medium-sized enterprises to take advantage of better information that enables them to run their businesses more efficiently (Vai, 2010).

In the health care industry, an attempt was made to develop and implement a BI solution to analyze the quality of life of patients with primary hyperhidrosis, before and after undergoing a video-assisted thoracic sympathectomy. The study concluded that the implementation of a BI solution in this field improved the analysis of the alterations in the patients' quality of life by allowing a better understanding of several factors (Gonçalves, Santos, & Cruz, 2011).

Another recent study intended to develop a BI solution for Portugal's intelligent transportation system that would support the companies' top management decisions. The authors of the study concluded that the use of the proposed BI solution would help decision makers in the transportation sector (Sampaio, et al., 2011).

2.2 Mobile business intelligence

As seen, the importance of BI is evident in the various industries, with companies seeking better ways to use the available information to their advantage. In addition, the shift from a wired to a wireless world of connectivity has placed handheld devices like smartphones and tablets on the center of a new era of mobile computing (Ramakrishnan, 2008). The idea of convergence between BI and mobility is obvious within this context (Airinei & Homocianu, 2010) with executives and information consumers starting to realize that they can easily and effectively analyze the same information on their smartphone and tablet devices as they did on their desktop computer or laptop in their office. The advantages of consuming mobile BI being that informed decisions can be made in the field in real time, instead of being delayed until a later date (Capgemini, 2011).

A recent study, assessing the current state of mobile BI as well as its future, states that mid-sized BI vendors are presenting advanced, yet very usable mobile BI solutions. All of these solutions support the iOS® platform, 90% support Android, 50% support BlackBerry® and approximately 30% support Windows® (Tabbitt, 2013).

Therefore, companies are focusing their attentions on these devices as a way to equip their employees with tools capable of increasing their productivity. A market study, assessing the changes regarding mobile BI in many industries, has verified that 68% of the companies consider the existence of mobile BI as "critical" or "very important" (Dresner, 2011).

2.3 User-centered design

In order to develop applications capable of delivering an intuitive BI experience, the most important thing to do is to center all the development activities around the target user – User-

Centered Design. The role of the UCD method is vital to the success of site and/or application development (Battilana, 2008).

Concerning the UCD, it is a structured methodology that involves users in all of their activities since the early stages of the development process (Nunes, 2006) and it is widely considered the key to product usability and usefulness (Mao, et al., 2005). The UCD methodology is standardized in ISO 13407 (1999), which describes the four essential UCD activities as follows:

- “Understand and specify the context of use”;
- “Specify the user and organizational requirements”;
- “Produce design solutions”;
- “Evaluate designs against requirements”.

The way these activities are connected is shown in Figure 2.1.

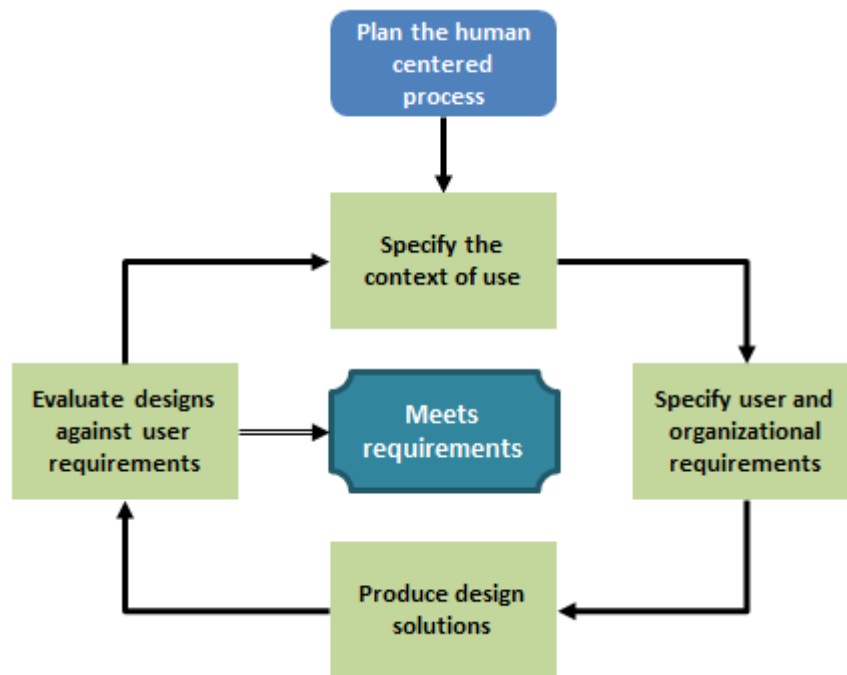


Figure 2.1 – The activities of the UCD according to ISO 13407 (1999).

Based on the UCD principles, a method called Schaffer-Weinschenk™ was developed to optimize user experience and performance, its steps are described as follows (Navalkar, 2004):

1. **Plan the project** – Identify the main activities and adequate staff and time. Establish the extent of usability work required;
2. **Evaluate the current applications** – Identify potential improvements;
3. **Know what the organization wants** – Identify business rules and directions before actual design;

4. **Know what the users want** – Understand different users and their needs to help build a solid foundation;
5. **Design the structure** – Ensure that users can understand what is offered, find things quickly, and navigate efficiently;
6. **Check standards** – Use standards to save time, improve design quality, provide consistency and help concentrate creativity;
7. **Design screens** – Create screen designs once the navigation and standard templates are in place;
8. **Support implementation** – Hand over the functional specification to the implementation team;
9. **Evaluate usability** – Complete a full simulation test. Maintain an ongoing process of monitoring site performance;
10. **Localize the application** – Create localized versions of the design to accommodate additional cultural contexts if required.

The following paragraphs list several examples of the importance of implementing UCD in the developments of products from various industries.

Early uses of UCD are present in the development of a Geographic Information System easier to understand, learn and use (Lanter & Essinger, 1991).

In the videogame industry, a study was performed attempting to address the challenges in game design by following UCD methods (Pagulayan, et al., 2002). The videogame industry is a good example that reflects the importance of following the UCD approach because users do not want to read manuals to play games (Kent, 2000).

More recently, in the military industry, UCD was successfully followed in the design of the graphical user interfaces (GUI) of advanced communication devices used in challenging environments (Luostarinen, et al., 2010).

In the health care industry, an attempt was made to follow the UCD approach to develop a technological platform aiming to aid diabetes' patients in managing their disease. In this study, the user's needs were collected through interviews and meetings, and then mockups of the platform were prepared and validated through heuristic evaluation. Finally, usability tests were carried out to find usability problems (Fico, et al., 2011).

In the retail industry, in order to improve customers' experience and generate excitement, the UCD approach was used to develop an interactive shopping window prototype that focuses on the information needs of teenagers. The user's needs were collected by observing customers behavior

and conducting interviews. Next, a prototype was created and usability testing was performed in the early stages before creating a fully interactive shopping window after analyzing existing similar products. (Zagel & Bodendorf, 2012).

As said before, the UCD approach can be successfully applied to products from various industries and contexts. Concerning mobile applications, a company called Digia was among the first to develop applications for smartphones. The first two applications developed by Digia to include the UCD approach were a global positioning system navigator and an image edition application. After their initial experience with the UCD approach, the company's workers came to the conclusion that UCD practices help to ensure the product operates properly (Kangas & Kinnunen, 2005).

Regarding the development of mobile applications nowadays, they are usually linked with touchscreen technology. The use of touchscreens has grown in the past decade and it has become increasingly common to have them integrated with handheld mobile devices (Birch, 2012).

A good example of designing for touchscreens can be found in public kiosks. The University of Pittsburgh has developed a public touchscreen display system for the students of its School of Information Sciences in order to provide them useful information. Its development followed the UCD approach and involved the collection of user's needs, prototype development and usability testing (Zhang & Jeng, 2011).

Also in the context of designing for touchscreens following the UCD method, a touchscreen prototype of a point of sales application was developed. The project involved creating a set of guidelines for touchscreen point of sales applications which were later applied in the design of the application GUI (Sjöberg, 2005).

Reducing the scope of the UCD implementation to handheld mobile devices supporting touchscreen technology, a few studies can be found in the literature.

In the health care industry, the UCD approach was followed to develop a tablet application as a new method of collecting additional data for the skin cancer research from the patients in the hospital. The outcome of the project was very positive, with patients being satisfied with the application, medical professionals saving up to 90% of time and hospital managers saving money (Holzinger, et al., 2011).

Another study, with the purpose of developing an on-device portal prototype for a touchscreen mobile phone, incorporated UCD activities by investigating the target user and the on-device portal context, designing the prototype iteratively and evaluating its usability with a target user focus group (Björnskiöld & Johansson, 2008).

There is another study which ran through the various stages of the UCD methodology in a similar manner as the one presented in this dissertation. Such study involved the development of a mobile application for the elderly tell their stories. In this study, the UCD was used to focus on the elderly's specificities in order to develop an adequate product, easily used by them. The various stages of the UCD present in this study contained user research, user's needs collection, low-fidelity prototypes design and also usability tests (Tenreiro, 2011).

A prime example of the importance of following the UCD methods in the development of a mobile BI application can be found in a real-world case study, in which a company was in charge of developing a mobile BI application for a major consumer products company in North America. This BI application was initially destined to the top fifty executives and vice-presidents of the company, who were seeking to gain mobile access to unit sales volume data. The application made such an impact that the company immediately decided to expand the implementation of the BI application to other business units. One of the factors responsible for such a success was the inclusion of users in the early stages of product development, with users actively participating in workshops where they could give feedback on the designs and help improving them (Capgemini, 2011).

2.4 Usability testing

One of the most important factors of the UCD approach is usability. ISO 9241-11 (1998) defines usability as the "extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use". More recently, ISO/IEC 9126-1 (2000) defines usability as "the capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions".

There are several methods that contribute to ensure the usability of a product. Most of these methods are applied in a specific phase of the development of a product (designing, prototyping, testing and evaluation) (Simões-Marques & Nunes, 2012). The importance of integrating the concept of usability in the development of products takes shape in activities such as heuristic evaluation and usability testing.

Heuristic evaluation

One of the main methods applied while evaluating usability of an interface is the heuristic evaluation. This method consists in inspecting the interface and comparing it against a set of recognized heuristics/principles/guidelines. The inspection of the interfaces should be done by a group of at least three usability experts (Simões-Marques & Nunes, 2012).

Heuristic evaluation was the most popular UCD approach in the 90s. However, nowadays it has become less relevant due to the growth of the mobile market and consequent departure from desktop applications (Cockton, 2013). Still, it's an important activity in the product development cycle.

Some of the most well-known and used heuristics are the following:

- **Nielsen's 10 Usability Heuristics for User Interface Design**, which, as the name suggests, consists in a set of ten general principles for interaction design (Nielsen, 1995);
- **Connell's Usability Evaluation Principles**, which consists in a set of thirty principles grouped into five categories (Connell, 2000);
- **Nielsen-Shneiderman Heuristics**, which consists in a set of fourteen principles based on Nielsen's heuristics and Shneiderman's golden rules (Zhang, et al., 2003);
- **Shneiderman's Eight Golden Rules of Interface Design**, is a set of eight principles applicable in most interactive systems (Shneiderman & Plaisant, 2005);
- **Weinschenk and Barker's classification**, which consists in a list of twenty principles collected from various sources (Sauro, 2011).

There are a lot of examples in the literature regarding the heuristic evaluation of various systems, some are depicted next.

A study, aiming to evaluate the design of a tailored system for self-care management of depressive symptoms in people living with HIV/AIDS, compared it against Nielsen's 10 Heuristics and helped the authors find some serious usability issues (Lai, 2007).

Another study, also in the health care industry, used Nielsen-Shneiderman Heuristics to conduct a heuristic evaluation to evaluate the treatment delivery system at a hospital in Toronto, Canada; and concluded that heuristic evaluation is an efficient method that can be successfully applied to radiotherapy delivery systems to detect usability issues and improve patient safety. The heuristic evaluation resulted in the identification of 75 usability issues (Chan, et al., 2012).

Another study adapted and tailored Nielsen's 10 Heuristics in order to evaluate a website for low-literate parents in the Neonatal Intensive Care Unit. The results from the evaluation allowed the authors to categorize and prioritize the usability problems which, in turn, eased the design of modifications to the website (Choi & Bakken, 2010).

A study, aiming to examine the evolution of security-related features of software like Microsoft® Internet Explorer® and Word®, used Nielsen's heuristics as the basis for a summary-level evaluation. The author of the study concluded that the security-related features had some usability problems which the heuristic evaluation helped detecting (Furnell, 2007).

Cognitive walkthrough

While heuristic evaluation is considered an informal usability inspection technique (Choi & Bakken, 2010) and favor usability experts over representative users, in formal usability testing, representative users are observed using a site or prototype while performing a set of typical user tasks in order to achieve predefined goals (Battleson, Booth, & Weintrop, 2001). This is procedure is commonly known as Cognitive Walkthrough.

The output of the Cognitive Walkthrough can take many forms, depending on the goals of study, the product tested and the goals of the users. Concerning the goals of the study, these can either be of a formative or summative nature (Tullis & Albert, 2008).

A formative study involves periodically evaluating a product or design, identifying insufficiencies, making recommendations and improving the design according to the recommendations. This process is repeated until the product is released, ideally, close to perfection. On the other hand, a summative study involves assessing how well a released product meets its objectives and perhaps compares it to other similar products. While formative testing focuses on identifying ways of improving products before their release, summative testing aims to evaluate a product against a certain set of criteria (Tullis & Albert, 2008).

Regarding the user's goals, depending on what they are, two main aspects of the user experience should be measured: performance and satisfaction. Performance metrics tell many things about how the user interacts with the product. It includes measuring task success, task time, the amount of effort to accomplish each task, the number of errors committed, etc. These metrics are critical for many products, especially those destined to serve professional purposes (such as company's internal applications). On the other hand, satisfaction metrics deal with the user's comments and thoughts about his/her experience while interacting with the product (Tullis & Albert, 2008).

While performance metrics are collected throughout the usability test, satisfaction metrics can also be collected after the tasks (post-task ratings) or after the entire testing session (post-session ratings). Satisfaction metrics are self-reported and can be collected through various methods like open-ended questions or questionnaires. The methods to collect post-task ratings include the following (Tullis & Albert, 2008):

- **Ease of Use**, which consists in a single question that asks users to rate the how easy/difficult each task was;
- **After-Scenario Questionnaire**, which is a set of three questions that touch the three fundamental areas of usability (effectiveness, efficiency and satisfaction);

- **Expectation Measure**, which consists in two questions, one answered before attempting the task and the other after finishing it. The “before” rating is called the expectation rating while the “after” rating is called the experience rating. Both rating are then displayed in a scatterplot;
- **Usability Magnitude Estimation**, which takes a slightly different approach on post-task ratings. This method consists in asking participants to assign “usability values” for each task as they complete them, gradually building a “usability ruler”.

An online usability study, aiming to compare several methods of collecting post-task ratings, asked 1,131 participants to perform six tasks on a website. The results of the study showed that the pattern of the results was very similar for all methods, which wasn't a surprise due to the large size of the sample. Then, the results of a smaller sample were analyzed and the conclusion was that the “Ease of Use” method was the most appropriate for smaller sample sizes (Tedesco & Tullis, 2006).

Regarding post-session ratings, some common techniques to collect them are the following (Tullis & Albert, 2008):

- **Aggregating individual task ratings**, which consists in simply averaging the results of post-task ratings;
- **System Usability Scale**, which consists in a set of 10 statements to which participants rate their level of agreement through a 5-point scale;
- **Computer System Usability Questionnaire**, which is a questionnaire designed to be administered by mail or online. It consists in a set of 19 statements to which participants rate their level of agreement. The results can be divided in four categories (system usefulness, information quality, interface quality, and overall satisfaction);
- **Questionnaire for User Interface Satisfaction**, which consists in 27 9-point rating scales divided into five categories (overall reaction, screen, terminology/system information, learning, and system capabilities);
- **Usefulness, Satisfaction, and Ease of Use Questionnaire**, which consists in 30 statements to which participants rate their level of agreement through 7-point rating scales. The statements are divided into four categories (usefulness, satisfaction, ease of use, and ease of learning);
- **Product Reaction Cards**, which consists in presenting participants a set of 118 cards, each containing an adjective (e.g. intuitive, entertaining, stressful, usable, etc.). Some are worded positively while others negatively. Participants would then choose which cards they felt better described the system, pick the top five cards of their selection and explain why they chose each. It is a more qualitative approach.

An online study, intended to compare the several post-session questionnaires, measured the reactions of 123 participants while navigating in two web portals. Each participant would answer to only one questionnaire to evaluate both web portals. The authors of the study concluded that the System Usability Scale (SUS) produced the most reliable results across different sample sizes (8, 10, 12 and 14 participants) (Tullis & Stetson, 2004).

There are some examples in the literature regarding the implementation of these methods in usability testing.

An example can be found on the website of the University of Buffalo's libraries. The website was subjected to usability testing through a set of 11 representative questions aiming to determine whether or not the libraries' users could effectively use the website to perform specific tasks. The users' performance and comments proved invaluable in exposing many usability problems of the site and all parties involved were impressed by the effectiveness of usability testing (Battleson, Booth, & Weintrop, 2001).

Another website usability study, involved the application of the Cognitive Walkthrough method to evaluate the interface of a website destined for children aged between 5 and 7 years old. The study consisted in comparing the tests results of the children with adults by subjecting both groups to a set of representative tasks divided into several actions. The authors of the study concluded that the combined analysis allowed the discovery of a greater number of interface design problems, making the Cognitive Walkthrough a reliable source for finding usability problems present on an interface aimed at children (Mano & Campos, 2006).

A study involved the comparison of the usability of three different mobile phones when users attempt to shop, purchase and playback music and video media files on their mobile handsets. The study consisted in subjecting users to the Cognitive Walkthrough procedure. Users were asked to perform alternatively two sets of tasks (one for purchasing media and the other to playback the purchased media). Performance metrics were collected from the tests such as the time to complete the tasks, success rates and number of attempts as well as self-reported metrics such as ease of use and satisfaction. The results were then summed through the SUM (Single Usability Metric) in order to assess which tasks had a better performance in which mobile phone (Weiss & Whitby, 2008).

In sum, it has been seen that UCD activities like heuristic evaluation and the Cognitive Walkthrough are present in the development stages of products from various industries (including the retail industry). Both activities complement each other because they are equally efficient and effective in addressing different types of usability problems (Tan, Liu, & Bishu, 2009).

Literature review

This literature review presented an overall description of the main concepts addressed in this dissertation as well as some examples for each. More important concepts will be described and detailed as the activities/processes presented in this dissertation justify it.

3. Interface prototypes development

The process of developing any product capable of satisfying its target user's needs requires many steps that should be taken in a cautious, structured way. Generally, a product can only be positively accepted and used by the target user if it has usability. Ideally, usability engineering should accompany the entire lifecycle of the product, with important activities happening even before designing the actual user interface (Nielsen, 1993). The following chapter describes in great detail, every step taken in the process of developing the interface prototypes. From collecting the user's needs and "converting" them into system functional requirements, to finding appropriate software or understanding the differences between designing for different mobile operating systems, all the details are presented. Furthermore, several screen designs are displayed along with the detailed explanation of the reasons that led to such design.

3.1. User's needs assessment

Assessing the user's needs is a vital stage of the development of every product. Collecting the user's needs in the initial stages of the project and having them in mind throughout the entire process is crucial to obtain a product that provides good user experience (UX).

In order to develop an appropriate BI application capable of fully satisfying the user's needs, first and foremost it's absolutely necessary to answer two questions: "Who is going to use the product?" and "How is the product going to be used?".

To answer the first question, there are many ways which can help in building a profile of the target user. Market analysis or observational studies are usually conducted to characterize individual users (Nielsen, 1993). However, in this particular case, the target user is already defined – store managers in a retail company. Still, it's important to know the users' needs and be in touch with them.

The answer for the second question comes from a great understanding of the job position, the work environment as well as all the tasks performed on a daily basis.

In order to collect the user's needs, a sample of seven company's store managers was selected based on their availability and also their interest in this particular subject. All the store managers were male with an average age of 43 years (ranging between 34 and 52 years) and their working experience varied from 6 to 20 years, averaging at 13 years.

After having the sample defined, a set of individual interviews with each user was arranged. These interviews lasted for about 1 hour each and occurred in a casual way, within the users'

working hours and in their working place. The method used for these interviews varies based on the product in development, but their goal is always to find existing or latent needs (Kraft, 2012). The purpose of these interviews was also to present users with the project goals. These are the development of a BI application able to give users mobility and the ability to assist them in the decision-making process while they are away from their desks. With this thought in mind, the users were asked a few questions in the following structured way: (1) what Key Performance Indicators (KPI) they classified as important to have with them while they're working away from their offices, (2) which level of importance they assign to each one of those KPIs (Maximum, High or Medium), (3) what applications they currently access to retrieve the said KPIs and (4) which tasks they need to complete on a daily basis. All of their answers and additional comments were noted in paper and all the interviews ended with the promise of a second meeting, this time with all the users, in order to conclude this phase of the process.

After the interviews, the users gave a guided tour of their workplace while, at the same time, describing some tasks that usually arise on a daily basis. These tours were very important to gain a better understanding of the users' working environment, how the application would be used and also enabled to build a good knowledge of the company's processes. Also, empirical knowledge played an important part at this point. By observing users' routines and procedures it was possible to confront users with eventual needs that they didn't raise in the first place. For instance, a user that had a small notepad filled with memos was asked whether or not he felt the need for the BI application to have a feature that would ease that process, for example, a virtual notepad that would allow the user to write notes and send them via e-mail. The information noted from these tours was then added to the information collected from the interviews.

With the initial set of interviews and tours concluded, it was possible to know the users' needs as well as understand what kind of usage would be given to the BI application in question. A total of 29 needs were collected from the seven interviewed users as shown in Figure 3.1.

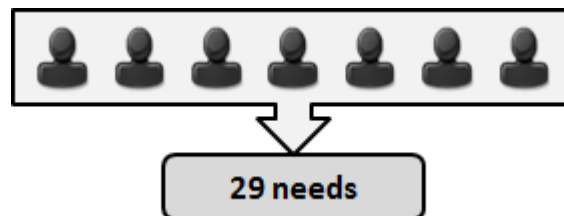


Figure 3.1 – Seven users raised 29 needs.

The next step taken was to list, in a structured way, the 29 user's needs collected from the interviews. The needs were codified with a number from 1 to 29 and organized in a table with four columns: (1) Code, (2) Need description, (3) Level of importance and (4) Amount of users

that raised the need. This table was built with the help of software such as Microsoft Excel[®]. A fraction of the table (with five examples) is presented in Table 3.1.

Table 3.1 – Fraction of the organized list of collected user’s needs.

Code	User’s need description	Level of importance	Amount of users
1	Analyze KPI 1	Maximum	7
11	Access the information anywhere outside the workplace	Maximum	5
12	Analyze KPI 8	High	3
20	Send and receive e-mails	Maximum	7
26	Access the workplace’s video feed	Medium	3

Having the needs organized in this structured way allows for a better future prioritization because the user’s needs could easily be arranged by different criteria such as level of importance or number of users raising each need (more on the prioritization process in subchapter 3.2).

The following step consisted in analyzing and listing the currently used BI applications in a similar organized way because a large part of this job involved analyzing KPIs which are spread across many BI applications available both on the computer and the smartphone. Such analysis is very important because users are already experienced with these applications and can point some user experience problems or simply indicate some improvement opportunities. Due to these applications’ massive nature (displaying a great amount of data), they can’t simply be ported onto tablets and smartphones mainly due to screen size restrictions. This situation leads inevitably to an overhaul in terms of the interface design of the developed BI application.

At this point, the KPIs classified as important by the users (regardless of the importance level attributed to them) were crossed with the currently used BI applications. The result, as shown in Table 3.2, is the list of currently available BI applications that give access to the KPIs classified as important. The applications were classified by name, platform availability (Computer, Smartphone or EDA – Enterprise Digital Assistant) and type of KPIs accessible from them. The applications were codified as letters A to F and the indicators as numbers 1 to 10.

The purpose of this procedure was assess the sources of information in order to narrow down the amount of applications that were going to be subject of a future analysis. Such analysis aimed to understand the way the applications were used, their design and information architecture and which fragments of their content/functions can be considered irrelevant for the scope of the developed BI application.

Table 3.2 – List of currently used BI applications and the KPIs accessible from them.

Application Name	Platform Availability	Key Performance Indicators
A	Smartphone (iOS)	KPI 1, KPI 4
B	EDA	KPI 3, KPI 9
C	Computer (Windows)	KPI 1, KPI 2, KPI 3
D	Computer (Windows)	KPI 6, KPI 7, KPI 8, KPI 9
E	Computer (Windows)	KPI 4
F	Computer (Windows)	KPI 1, KPI 2, KPI 3, KPI 4, KPI 5

As seen in Table 3.2, there are six applications currently used by the store managers, which contain KPIs considered important for the scope of the BI application in development. Out of these applications, two of them are considered mobile (A and B). Application A refers to a smartphone application available only for iOS (Apple mobile operating system) while application B refers to an EDA which is a mobile device similar to a Personal Digital Assistant (PDA) but with a more rugged build (Janssen, 2012). The remaining four applications are software for Windows and the users have access to them in their office's computer. The KPIs available in all of these applications range from sales performance (KPI 1) to the size of the waiting lines (KPI 6).

Due to sharing the mobility feature with the mobile BI application in development, the currently available mobile BI applications were then subject of a more detailed analysis. This analysis intended to justify the need to develop the new mobile BI application by pointing the gaps of the available content of the current mobile BI applications. With this in mind, it's clear that both applications A and B do not provide the access to every KPI considered important for the users (only KPI 1, 3 and 10 in application A and KPI 4 in application B are available) and so, the need to develop a new mobile BI application was completely justified and plausible.

And so, the first phase ended and it was time to prioritize the user's needs collected at that point.

3.2. System functional requirements definition

After collecting, listing and organizing all the user's needs, it was important to have them presented in a certain priority order. The prioritization was going to be defined according to certain criteria so the most important needs would be ranked at the top level of a list while the not so important ones would be placed at the bottom level of that list. It was decided that the user's needs were going to be prioritized according to the following criteria:

- Number of users raising each need;
- Level of importance of each need, assigned by the users (Maximum, High or Medium).

Having the needs listed in the way shown in Table 3.1 allowed to easily arrange them according to each individual criterion. This process is shown in Figure 3.2.

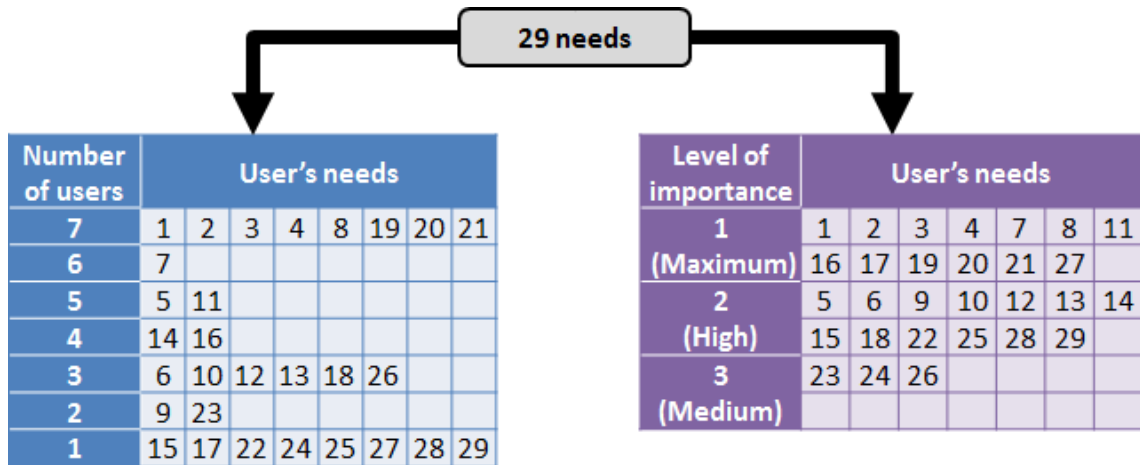


Figure 3.2 – User's needs prioritization criteria.

Having two separate tables, one for each criterion, was a good way to start the prioritization process and assess the importance of each user's need. Still, it was necessary to find a way of combining both criteria in order to have only one list of prioritized needs. The solution found was the creation of four levels of prioritization (A, B, C and D), as depicted in Figure 3.3.

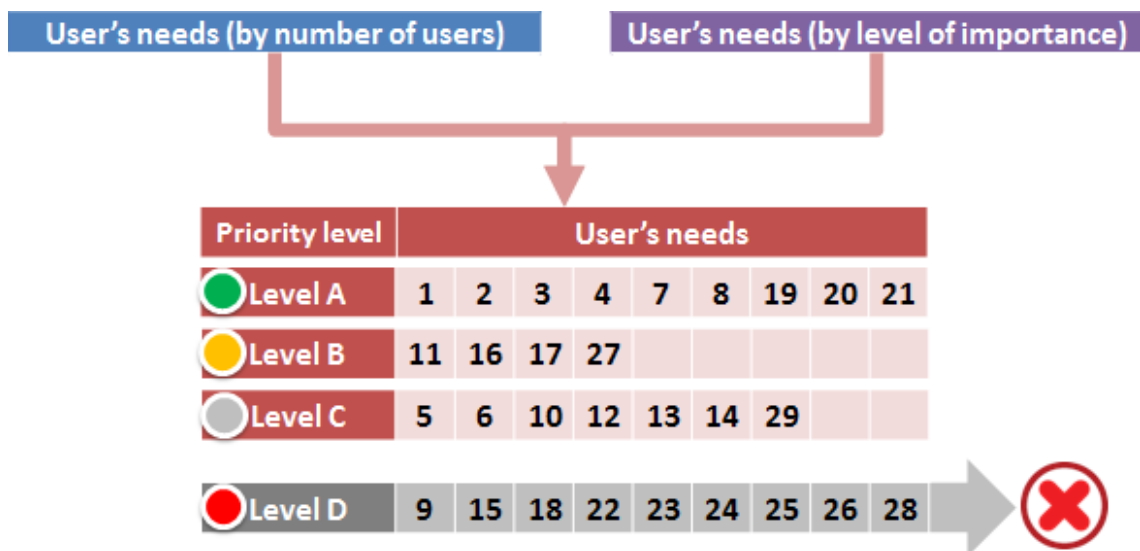


Figure 3.3 – Final prioritization of the user's needs.

The criteria used to group the needs on each level were the following:

- Level A – User’s needs classified with a maximum level of importance, raised by at least 80% of the users (e.g. all seven users raised the need number 20, and, at the same time, classified it with maximum importance);
- Level B – User’s needs classified with a maximum level of importance, raised by less than 80% of the users (e.g. four users raised the need number 16 and classified it with maximum importance);
- Level C – User’s needs classified with a high level of importance, raised by any number of users and considered to be of reasonable implementation (e.g. five users raised the need number 5 and classified it with high importance);
- Level D – Discarded user’s needs. Needs classified with a high/medium level of importance, raised by less than 50% of the users and considered to be of not so reasonable implementation. For instance, only three users raised need 26 and classified it with medium importance. This need refers to the ability of viewing the workplace’s closed-circuit television from the smartphone/tablet and the possible future benefits from its implementation wouldn’t be proportional to the probable high costs of implementing such functionality

As seen in Figure 3.3, the list of needs was shortened from 29 to 20 needs due to having 9 needs assigned to level D and thus, discarded. The remaining 20 needs were distributed by the other levels (nine needs in level A, four needs in level B and seven needs in level C) according to the criteria described earlier.

After having the user’s needs prioritization defined, the next step consisted in performing an analysis which placed side by side, the current constraints of not having the need satisfied by a mobile BI application available versus the future benefits of having the need satisfied by a mobile BI application.

This analysis was performed for each one of the 20 user’s needs from levels A, B and C. Such analysis is very important because the weaknesses identified in the current situation may present opportunities for future improvements (Nielsen, 1993). For example, Table 3.3 lists the current constraints of not having KPI 1 (with the desired level of detail) available on a mobile BI application as well as the future benefits of having KPI 1 available on the mobile BI application in development.

As it was referred, for each current constraint, the future benefit was presented. For user’s need 1 (Analyze KPI 1), currently the store managers had to travel to their office, and either analyze KPI 1 directly on the computer at their desk or select the desired information and print it to analyze it somewhere else. Alternatively, they could use application A on their smartphone and perform the

analysis. However, application A doesn't grant KPI 1 the level of detail desired by the store managers to make a proper analysis. These are the current constraints listed in Table 3.3. On the other hand, if the mobile BI application in development was available to use, the benefits of this scenario would be that the store managers wouldn't need to travel to their office and analyze KPI 1 on their desktop or print anything because they would be able to make the desired analysis of KPI 1 anywhere, using their smartphone/tablet.

Table 3.3 – Current constraints vs. future benefits analysis for user's need 1.

User's need 1 – Analyze KPI 1	
Current constraints	Future benefits
Time spent by the worker in travelling to the office and selecting the desired information on the computer.	Time saved by the worker in being able to analyze the information everywhere.
Paper and ink wasted in printing the information.	Paper and ink saved.
Desired level of detail for KPI 1 isn't attained by application A.	Desired level of detail for KPI 1 would allow the worker to act faster.

Because of the wide variety of current constraints and possible future benefits which arose from the several user's needs gathered in Phase 1, one example may not be enough to completely illustrate the importance of such analysis. Therefore, two more examples regarding the analysis performed for needs 11 and 19 are presented in Table 3.4 and Table 3.5, respectively.

Table 3.4 – Current constraints vs. future benefits analysis for user's need 11.

User's need 11 – Access the information anywhere outside the workplace	
Current constraints	Future benefits
Inability to access certain content/applications outside of the workplace.	Application and its content available everywhere.
Delay in the treatment of some matters.	Faster acting in the treatment of some matters.

Regarding user's need 11, currently the store managers can't access most of the applications outside of their workplace which, in turn, causes some delays in the treatment of some matters. If

the mobile BI application were available, the store managers could access the desired content anywhere, increasing their productivity because they would act faster on the matters.

Table 3.5 – Current constraints vs. future benefits analysis for user’s need 19.

User’s need 19 – Register notes and send them by e-mail	
Current constraints	Future benefits
Time spent in manually taking notes and later entering them on the computer to send by e-mail.	Notes are taken directly on the devices and sent by e-mail sooner, increasing the productivity of the worker.
Time spent in manually taking notes to deal with issues later because the information needed to solve the issues is only available on the computer.	Store managers do not have to take notes because they have the chance to deal with the issues at the time they emerge.

This last example refers to the user’s need of having the ability to take notes (written, audio or video) and send them by e-mail. Currently, the store managers usually carry a small notebook and write notes of things/situations they consider important and need to either analyze in more detail later, or send by e-mail to fellow co-workers. The time spent in taking the notes and entering them again on the computer could be used for something else, more productive, if they had the ability of taking notes directly on their mobile devices and send them by e-mail. Also, if the information is available on the BI application in development, instead of taking notes of something they need to analyze later on the computer, they could analyze it right away by accessing the application. Altogether, the store managers would have an increase in productivity.

After characterizing both current constraints and future benefits for each of the 20 user’s needs from levels A, B and C, the next step involved “converting” the user’s needs into detailed system functional requirements. In other words, the needs describe the information content whereas the functional requirements give details on how to provide that information in the application’s functional point of view. The purpose of this process is to aid in the future designing of the interfaces. Also, these functional requirements were accompanied by technical requirements, which will not be mentioned because they have no effect in the designing of the interfaces and thus, do not fall within the scope of this dissertation.

At this point it’s known what information users needed to analyze so, by “converting” the needs into system functional requirements, it was also explained how the users were going to be able to analyze the information (what they were going to see, which buttons they were going to be able to press, etc.). And so, a table was created for each of the 20 needs. Each table had 4 columns: (1)

Description, (2) Information, (3) Functionality and (4) Functional requirement. The process of creating the functional requirements for user's need 1 can be found in Table 3.6.

Table 3.6 – Converting user's need 1 into system functional requirements.

User's need 1 – Analyze KPI 1			
Description	Info. ¹	Func. ²	Functional Requirement
Access KPI 1 section		X	Button available in the main menu that allows the user to access the KPI 1 section of the app.
Select type of KPI 1		X	Menu with two options (“KPI 1a” and “KPI 1b”).
Table with 4 columns	X		After entering the desired KPI 1 section, a table with 4 columns is shown.
Column 1 (Store code & name)	X		The code and name of every store is listed in column 1.
Column 2 (KPI 1 value)	X		KPI 1 value for each store is listed in column 2.
Column 3 (ΔH)	X		The percentage of KPI 1 deviation relative to KPI 1 history (ΔH) is listed in column 3.
Column 4 (ΔO)	X		The percentage of KPI 1 deviation relative to KPI 1 budget (“orçado” in Portuguese) (ΔO) is listed in column 4.
Reference Date	X		The reference date is shown at the top of the table (previous day is shown by default).
Sort rows		X	Button available on the header of each column that allows the user to sort the information of the rows of that column in an ascending/descending order.
Access Chart		X	Button that allows the user to access the KPI 1 chart.
Refresh values		X	Button that allows the user to refresh the values.
Toggle between store group A, B or C		X	Segmented control that allows the user to toggle between store group A, B or C.
Return to the main menu		X	Button available on the top left corner of the screen that, when pressed, allows the user to return to the main menu of the app.

¹ Information

² Functionality

By detailing each feature of the interface in a logical way, the designing process would later be much simpler because every information/function was discriminated at this point. In the case of the user's need 1, the store managers intended to analyze the behavior of KPI 1 on the day before (by default). In order to perform this analysis they would access the KPI 1 section on the main menu of the BI application by pressing the respective button. Then, they would be presented with a table with the KPI 1 behavior in each store in the day before (KPI 1 value, KPI 1 Δ H and KPI 1 Δ O). The store managers could then analyze charts or sort the rows by KPI 1 value, Δ H or Δ O. This would be done by pressing the respective header button. Finally, a "back" button is available for the store managers to return to the main menu of the application. All of these steps can be clearly understood simply by consulting Table 3.6.

With every system functional requirement, current constraint and future benefit defined for each user's need, it was time to close the second phase by arranging a meeting with all the seven users. The purpose of this meeting was to validate the entire prioritization process along with each of the 20 user's needs (the discarded user's needs were no longer addressed).

The meeting took place in the company's headquarters. At the meeting, the users were greeted and seated around a table and the presentation started by reminding them about the purpose and importance of their presence. First, they were shown the entire prioritization process (how the criteria were defined and such) along with the list of needs comprising each priority level. After giving the overview of the process, and starting from priority level A, each need was individually analyzed. At that point, the users could participate with comments and exchange views with each other. After presenting each need, the users were asked whether or not they agreed with the way the need was defined as well as with the priority level assigned to it.

By promoting a group discussion where users could express their opinions while hearing others' opinions, it was almost guaranteed that every single decision regarding the content/functionalities to be included in the BI application would satisfy the majority. The outcome of the meeting was the following:

- Needs 11 and 17 were considered system requirements and should be prerequisites for the BI application in development and thus, removed from the list. User's need 11 refers to being able of using the mobile BI application anywhere, at any time. Being an application developed for mobile equipment like smartphones and tablets, this need should be considered a prerequisite. User's need 17 refers to being able to consult the same information regardless of platform and OS. Again, this should be considered a prerequisite because by designing the interfaces for both platforms (smartphone and tablet) and by following both iOS and Android

guidelines it's guaranteed that the same information will be available whichever platform and OS the store managers decide to use;

- Needs 19, 20 and 21 could be satisfied by smartphone/tablet native functionalities or by applications available for free on the market and thus, removed from the list. For instance, user's need 20 refers to being able to send and receive e-mails, which is a native functionality of every single smartphone and tablet. On the other hand, user's need 19 refers to the ability of register notes and send them by e-mail. There are many applications available on each OS' market that are able to satisfy this need;
- Needs 5, 6, 10 and 29 should be incorporated as part of need 16 and thus, removed from the list. Need 16 refers to the ability of receiving notifications about several occurrences while needs 5, 6, 10 and 29 refer to analyzing certain KPIs. These KPIs, although important, would be of greater value if converted and added to the range of notifications included in need 16;
- Needs 3, 8, 12, 13 and 14, although still considered important by the users, after discussing together, they agreed that each one of these needs falls out of the scope of mobility and had to be removed from the list. For example, user's need 3 refers to analyzing KPI 3 which is an indicator that's very important but requires a more careful and comprehensive analysis, which should be done behind a desk and not while moving around.

With this array of changes proposed by the users and approved by everyone, the list of user's need shortened from 20 to 6 needs. This reorganization of the list is summarized in Figure 3.4.

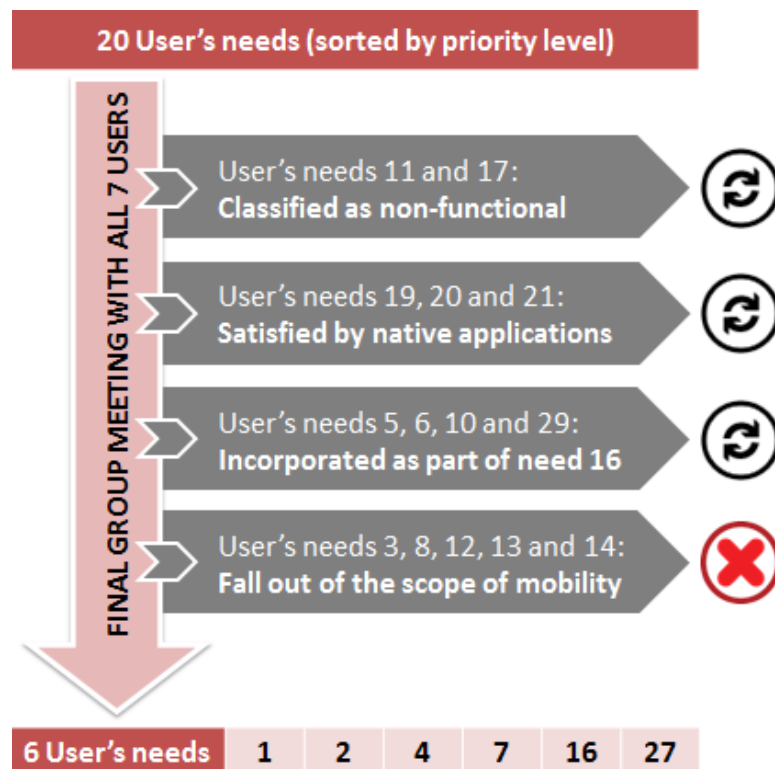


Figure 3.4 – Outcome of the final meeting.

Therefore, the final meeting resulted in a major reduction of the user's needs (14 needs were dropped from the list). This change may seem too drastic, however, only five needs were completely dropped while the others were just reclassified but will still be part of the mobile BI application in development.

The six user's needs remaining were the ones that were going to, effectively, require interface designing. They are as follows:

- User's need 1 - Analyze KPI 1;
- User's need 2 - Analyze KPI 2;
- User's need 4 - Analyze KPI 4;
- User's need 7 - Analyze KPI 7;
- User's need 16 - Receive notifications about several occurrences;
- User's need 27 - Access product's file.

After reaching an agreement, an explanation of the ensuing steps took place. The users were informed that afterwards they were going to take part in a process called Participatory Design in which more individual meetings were going to occur where they will have the opportunity to monitor and aid on the development of the interfaces prototypes. The extent of their role in the participatory design will be further discussed in subchapter 3.4.

This meeting concluded phase 2 of the development of the BI application. At that point, and from the company's point of view, every bit of information needed to start designing the interfaces was gathered and completely defined.

3.3. Preparations for the interface design

Before starting to design the two interface prototypes (smartphone and tablet), some things needed to be done first: research the market for similar BI applications, finding a suitable prototyping software, defining a prototyping approach and acknowledging the differences and similarities between iOS and Android user experience.

The purpose of the market research was to build up more knowledge on the specificities of the type of information that's going to be included on the application as well as gathering interesting ideas that may be adapted to the application (for example, various ways of presenting tables and charts).

Not all the researched BI applications were mobile; some were software available only for desktops or laptops. This fact did not pose a problem because, as said before, the idea at that point was to gather some inspiration and good design ideas, regardless of the platform of the

applications/software. However, in the case of the researched applications being available for smartphone and/or tablet, their design wasn't the only feature subject of analysis, their architecture was also targeted. In this type of applications it's very important to understand how the information is structured and how the user navigates between pages/sections of content. For instance, sliding between various charts instead of going back and select another chart makes the navigation faster and much more intuitive. This research turned out to be very fruitful, and some design and architecture ideas were collected from applications such as MicroStrategy^{®3}, Geckoboard⁴, Klipfolio^{TM5}, Roambi^{®6} and others.

Regarding the search of a suitable prototyping tool, having software with strong functionalities for developing the user interfaces, would ease and accelerate the prototyping process and increase the authenticity of the GUI and of the basic interaction functionalities. These facts would ultimately help the task of evaluating the usability of the application (Simões-Marques & Nunes, 2012).

And so, among all the software available, two options emerged: Prototyper and Mockups (both free editions) developed by Justinmind and Balsamiq[®] respectively. Both are software intended to build wireframes for mobile equipment. The choice fell on Prototyper because it provides a database of real iOS and Android icons, bars, buttons and several other interface elements for both smartphone and tablet, enabling a truthful recreation of the interface.

At this point it's important to emphasize that the purpose of this process is to design two interfaces prototypes, not the real interface. However, these interface prototypes are intended to represent an early statement of the kind of interface toward which the company is aiming. Having a truthful prototype makes the details of the design more evident for developers, encouraging them to follow similar principles in subsequent design activities (Nielsen, 1993).

The prototyping approach adopted was a mixture of Horizontal and Vertical prototyping.

Horizontal prototyping intends to simulate the entire user interface, but users cannot perform any real tasks on a system with little functionality. Its purpose is to assess how the interface feels as a whole. On the other hand, vertical prototyping aims to replicate a limited part of a full, functional system. It aims to test that part under realistic circumstances with real user tasks (Nielsen, 1993). Both of these approaches are illustrated in Figure 3.5.

By mixing both approaches, it was intended to replicate a scenario where the users could explore most features of the application without being able to interact with real data. It wasn't necessary

³ <http://www.microstrategy.pt/>

⁴ <http://www.geckoboard.com/>

⁵ <http://www.klipfolio.com/>

⁶ <http://www.roambi.com/>

to supply a large amount of content or enable every control in the prototype. However, it was important to provide the user with a realistic experience (Apple Inc., 2012).

The advantage behind having a scenario is that it can be used during the design of an interface in order to understand how the users will interact with the future system. It's also a good way to gather user feedback without the expense of constructing a fully functional prototype (Nielsen, 1993).

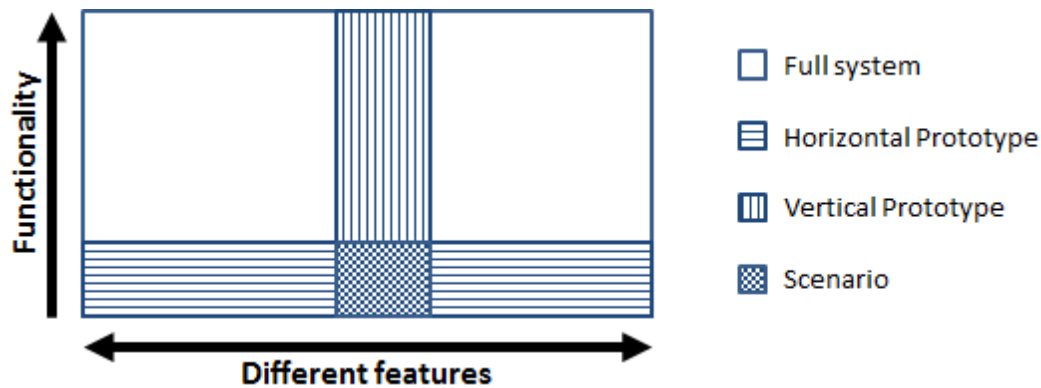


Figure 3.5 – The two approaches of prototyping (adapted from (Nielsen, 1993)).

After defining the prototyping approach, it's time to build knowledge on good practices for designing an application for smartphones and tablets. Apple and Android provide very good guidelines that intend to help designers and developers create applications that meet the respective companies' standards.

Android guidelines (called Android Design) recommend designers to consider three main goals while designing applications for Android. The first goal refers to enchanting the user by combining beauty, simplicity and purpose to create an effortless and powerful experience. The second goal relates to the simplification of the user's life by creating an application easy to understand where simple tasks never require complicated procedures while complex tasks are tailored to the human mind. Finally, the third goal deals with the user's satisfaction. This is attained by allowing users to use the applications in inventive new ways while, at the same time, giving the application a personal feel (Android, 2012).

Apple guidelines (called iOS Human Interface Guidelines) present six principles upon which designers should create their applications. The first principle refers to the application's aesthetic integrity which measures how well the appearance of the application is integrated with its function. The second principle relates to creating an application consistent with the iOS standards and itself. Such consistency allows users to transfer their skills and knowledge between applications. The third principle deals with granting users the ability to directly manipulate objects on screen instead of using separate controls to manipulate them. This principle aims to

make tasks more engaging while giving the users a greater sense of control over the objects on screen. The fourth principle involves having the application acknowledging user's actions by giving some sort of feedback such as subtle animations or notifications. The fifth principle of Apple human interface guidelines mentions the use of metaphors between virtual objects/actions in the applications with objects/actions in the real world. By allowing users to operate realistic on-screen objects as if they were real-world objects, the application becomes easier to understand. The sixth and final principle states that users should be the ones initiating and controlling the applications, not the other way around. Applications should find a correct balance between conceding users the capabilities they need while helping them avert possible dangerous outcomes (Apple Inc., 2012).

It's important to note that despite developing the mobile BI application for both operating systems (Android and iOS); only one interface was going to be designed for each platform (smartphone and tablet). In other words, the same interface is going to be presented to the users whether they access it on devices equipped with Android or iOS. The reason for this decision was mainly due to time constraints because designing different interfaces for each operating system would take almost twice the time (four interfaces would have to be designed instead of two). Besides, the purpose of designing an interface prototype was to show the interface's graphical elements placed in their rightful place and the architecture of the interface correctly assembled. The look of the graphical elements was of secondary concern.

However, it's still important to acknowledge the main similarities and difference between Android and iOS UX that need to be considered when designing applications.

The main similarities between Android and iOS applications are (Cornett, 2012):

- Overall structure – Both systems have a similar basic flow of information for their applications;
- Hierarchical navigation – Both systems have their applications structured in a similar hierarchical tree-based format, where the users can drill down into the information;
- Orientation – Both systems provide visible screen titles that indicate where the user is, where he came from and where can he go next;
- Expected functionality – Both systems have interface elements such as buttons, sliders, tabs and more that behave in a similar way;
- Gestures – Both systems support a wide range of gestures like tap, drag, swipe, etc.

Regarding the main differences between Android and iOS applications, they are as follows (Cornett, 2012):

- Screen sizes – iOS is only available for iPhone®/iPad® while Android is supported by many devices with different screen sizes and resolutions;
- “Back” navigation – Android devices have a physical “Back” button that goes back in screen history while iOS devices do not. Additionally, Android applications have an “Up” button that navigates back within an application. iOS applications have a “Back” button that has a similar function to Android “Up” button;
- Tab navigation placement: iOS has a five button (maximum) tab bar at the bottom of the screen while Android recommends that the tab bar (which is scrollable) should be placed at the top of the application. By having a scrollable tab bar, Android applications can have more tabs than those that can be displayed on a screen;
- Changing data views: For this functionality, iOS uses “segmented controls” while Android provides something called “spinner” which is a drop-down menu of options;
- Selecting from a list of actions: Android uses radio buttons listed on a dialog pop up box. On the other hand, iOS uses a control called “action sheet” to display a list of actions;
- Search: iOS provides a search bar at the top of a searchable screen. Android uses a search button that conceals the search bar.

Acknowledging the primary similarities and differences between both operating systems was a very important step towards bypassing the differences. Regarding screen sizes, the choice fell on using iPhone’s 320x480 and iPad’s 1024x768. About the “Back navigation”, this wasn’t really a problem because both iOS “Back” and Android “Up” buttons have the same function. Concerning tab navigation, the choice fell on iOS tab bar mainly because by looking back at the user’s needs that were going to be subject of interface design, one could group them into four different sections (KPIs, Notifications, Item file and Shortcuts for other applications) so the iOS approach seemed better. Regarding the changing views functionality, both versions were used. iOS segmented controls were used in some sections because it gave a bigger visibility to the options available. On the other hand, Android drop-down menu was used in other sections where the number of available options exceeded three and they didn’t need much visibility. Figure 3.6 illustrates this difference.

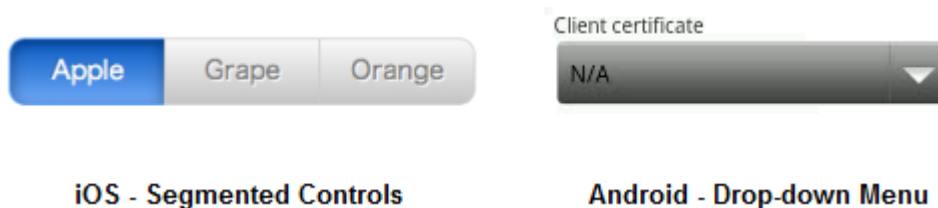


Figure 3.6 – Comparison between iOS segmented control and Android drop-down menu.

About the selection of lists of actions, it wasn't necessary to resort to this type of functionality, so this difference didn't pose a problem. The same happened to the search function.

In short, from an interface prototype point of view, the differences between both operating systems are somewhat significant but do not hurt the overall architecture of information of the application. In order to illustrate this concept, the similarities between the Facebook® applications for both systems are depicted in Figure 3.7. As observed, both interfaces are almost identical, having minor differences such as the search and chat buttons.



Figure 3.7 – Similarities between Facebook on iOS and Android.

So, the decision of designing only one interface for each platform (smartphone and tablet) prevailed. At this point, another decision was made towards the way of designing the interface prototypes. Since the Prototyper software offered a library GUI elements for both iOS and Android, there were four options available: Use only iOS GUI elements, use only Android GUI elements, use a mixture of both or ignore the software's library and create the GUI from scratch. Being a hybrid of both OS, the choice was to use mainly iOS GUI elements, resorting to Android GUI elements whenever appropriate.

Both prototypes were going to be designed in parallel, that is, whenever a section of the smartphone interface was designed, that same section would be designed for the tablet interface. The smartphone interface sections were designed first because it seemed more natural to expand the smartphone design to the tablet rather than transitioning the tablet design to the smartphone.

Also, smartphone and tablet users expect their applications to rotate along with their devices. However, due to time constraints, the interface prototypes design was restricted to one orientation for each interface. The choice fell on the more natural position for each: the smartphone interfaces were going to be designed in portrait mode (with charts being presented in landscape) and the tablet interfaces were going to be designed in landscape mode. Concerning tablets orientation, a study targeting the way the top applications are presented in Apple App StoreSM (screenshots in portrait or landscape mode) revealed that, when it comes to the top financial and business applications available in the market, they are mostly advertised with screenshots in landscape mode (Park, 2011). As shown in Figure 3.8, only around 10% of the finance applications are advertised exclusively in portrait mode while 60% are presented exclusively in landscape mode. Regarding business applications, the percentage of applications presented exclusively in portrait mode is even smaller (around 8%), with another 8% being presented mostly in portrait mode. Of course, these numbers do not reflect how people will actually orientate their devices when using finance and business applications but gives a good perception of how these applications are designed.

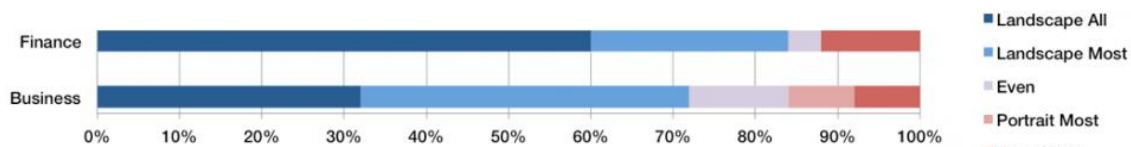


Figure 3.8 – Tablet preferred orientation in finance and business applications advertising (adapted from (Park, 2011)).

Arriving at the moment of starting the design of the interfaces, it is important to enumerate the various factors that were going to contribute to the design:

- 6 user's needs required interface design;
- Two interfaces (smartphone and tablet) were going to be designed;
- Justinmind's Prototyper software was selected to design the interfaces;
- BI applications such as MicroStrategy, Roambi, and others provided a collection of good design ideas;
- The prototyping approach chosen was a mixture between Horizontal and Vertical Prototyping – Scenario;
- The primary similarities and differences between iOS and Android UX were acknowledged;
- iOS GUI elements were going to be used predominantly, with Android GUI elements being used whenever appropriate;
- The interfaces were going to be designed in parallel with the smartphone interface designed first and then expanded for the tablet prototype;

- The smartphone interfaces were going to be designed in portrait mode (with the exception of the charts) and the tablet interfaces in landscape mode.

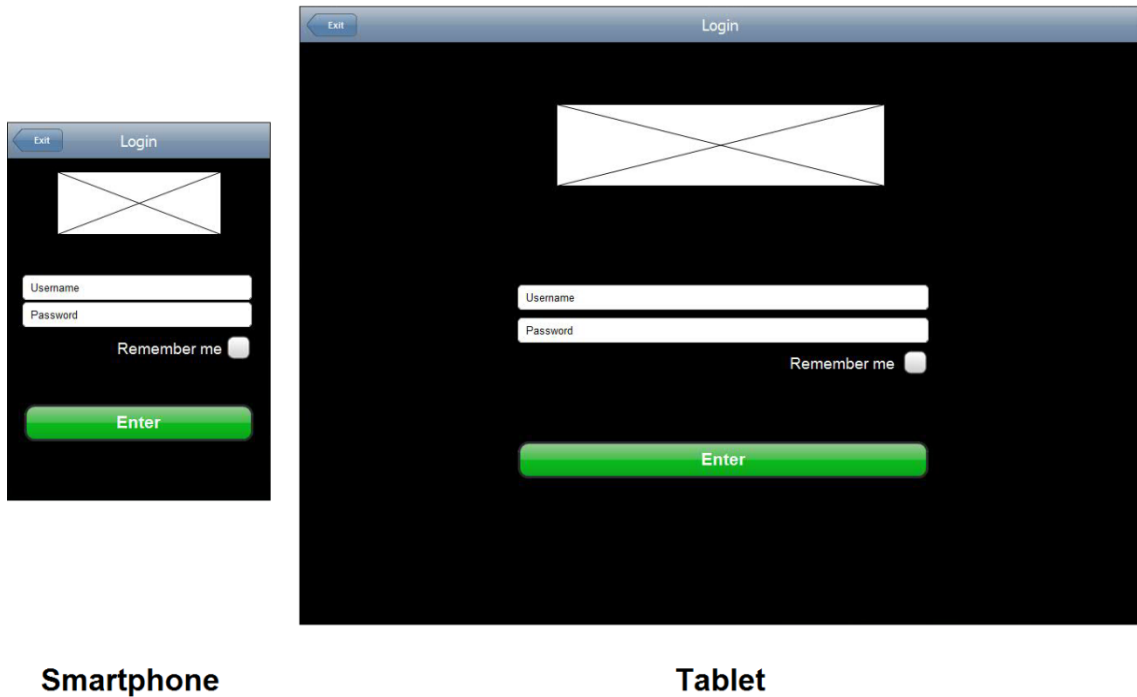
3.4. Interface design

It's now time to describe how the main screens for each section were designed along with the rationale used while designing them. Note that the terminology present in the interfaces, although mostly concealed in this dissertation, was originally presented in Portuguese and later translated to English for a better integration in this dissertation. All the system functional requirements defined in Phase 2 (see subchapter 3.2) were of extreme importance at this point because they worked almost like guidelines, listing all the functionalities and information that would populate each screen.

There were two criteria behind the choice of which screens were going to be designed. First of all, there was the need to showcase all the different functionalities and sections of the application. In addition, by following a scenario approach, several more screens were designed to later provide testing subjects with logical paths to follow in the Cognitive Walkthrough from Phase 4 (detailed in subchapter 4.3).

The first screen designed was the "Login" screen. Being a BI application intended to serve the store managers of a company, a login was mandatory to prevent unwanted people from accessing the application. The login screen concept is very simple; users just type in their username and password and press the green "Enter" button. Users can also select a box if they wish the application to remember their credentials. Ideally, the application should provide users a way to recover their password if they happen to lose/forget it. However, in this case the credentials entered in the "Login" screen aren't exclusive to the mobile BI application, therefore the whole credentials recovering process is taken somewhere else. Finally, there's an "Exit" button on the top left corner of the screen. Users can press that button to exit the application or simply press the "Home" button in the physical device. Both prototypes have similar screens as depicted in Figure 3.9.

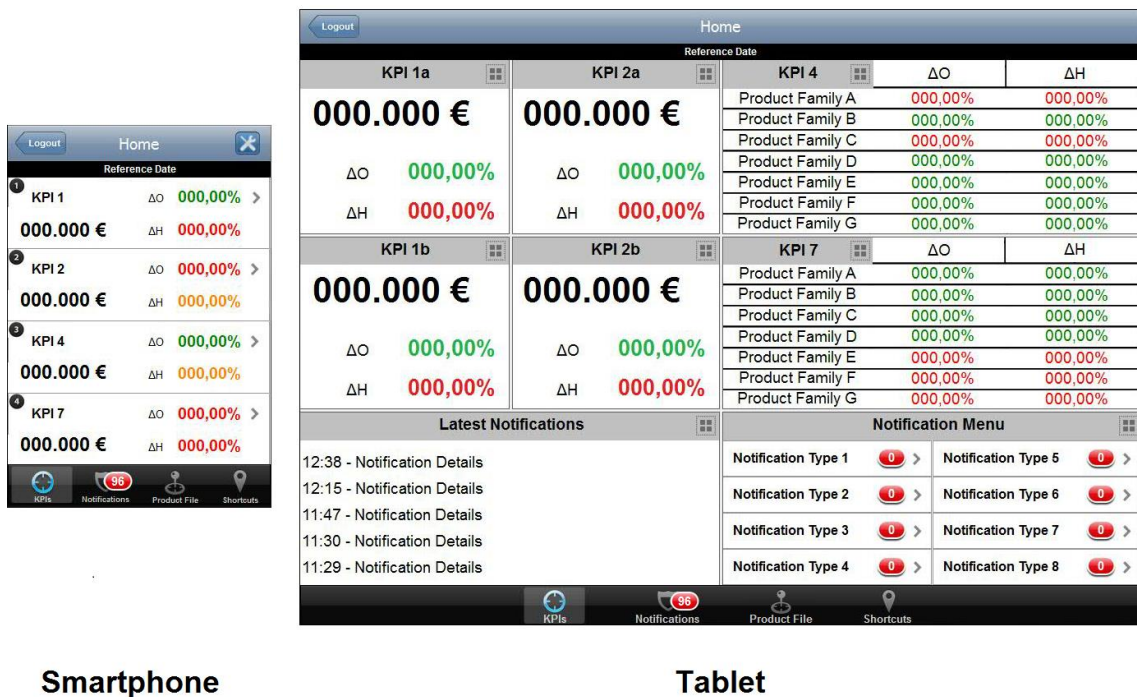
The next screen designed was the main menu screen (known as "Home"). This screen differs a little bit from normal menu screens because it doesn't present users with just the names of the different sections contained in the application like the usual main menus do. Instead, it works as a dashboard that gives users an overview of the KPI's behavior as soon as they enter the main menu. Figure 3.10 depicts both versions of the "Home" screen.



Smartphone

Tablet

Figure 3.9 – “Login” screen comparison between the smartphone and tablet interfaces.



Smartphone

Tablet

Figure 3.10 – “Home” screen comparison between the smartphone and tablet interfaces.

Both “Home” screens are divided into several sections. The smartphone version has four sections (one for each KPI) while the tablet version has eight. Naturally, the larger screen dimensions of

the tablet allow more information being presented on screen. However, every piece of information on the “Home” screen of the tablet is also available on the smartphone version.

Both interfaces present the various KPIs (1, 2, 4 and 7) corresponding to user’s needs 1, 2, 4 and 7, respectively. In addition to presenting the KPIs’ behavior on the reference date (always the day before), each section also works as button that can be pressed in order to take the users further into the detail of that KPI. The colors of the values intend to classify their performance on the reference date, green means “good”, orange means “normal” and red means “bad”. Although this isn’t an actual graphical element of the interface, its importance was emphasized at this point in order to be taken in consideration when the actual technical implementation occurs. Having the numbers in such colors helps users identifying which KPIs should be target of a more comprehensive analysis.

Following iOS guidelines, there is the navigation bar at the top, displaying the title of the screen, centered along its width. In the smartphone version, there is a “Configurations” button at the top right corner of the screen. The purpose of this button is for users to customize the “Home” screen by selecting which information of each KPI they wish to have on the four sections of the main screen. The tablet version doesn’t have the “Configurations” button. However, by pressing the grey button on each section’s header, users summon a popup with a similar set of options as the smartphone version. Also, in the tablet version users can move each section to a different position by dragging it around on the “Home” screen.

On the bottom of the screen there is a tab bar which gives users the ability to change between the several areas of content. The first area refers to the KPIs and by pressing it, users are directed to the “Home” screen. The other areas are “Notifications”, “Product’s File” and “Shortcuts” which will be approached later in this subchapter. The “KPIs” area attends to user’s needs 1, 2, 4 and 7, the “Notifications” area covers user’s need 16, “Product’s File” area covers user’s need 27 and lastly, the “Shortcuts” area allows users to add shortcuts to other applications such as internet navigator, e-mail, contacts, notepad and such. The creation of the shortcuts area intends to satisfy the user’s needs that were labeled as “satisfied by native applications” after the final meeting with all the users (see subchapter 3.2).

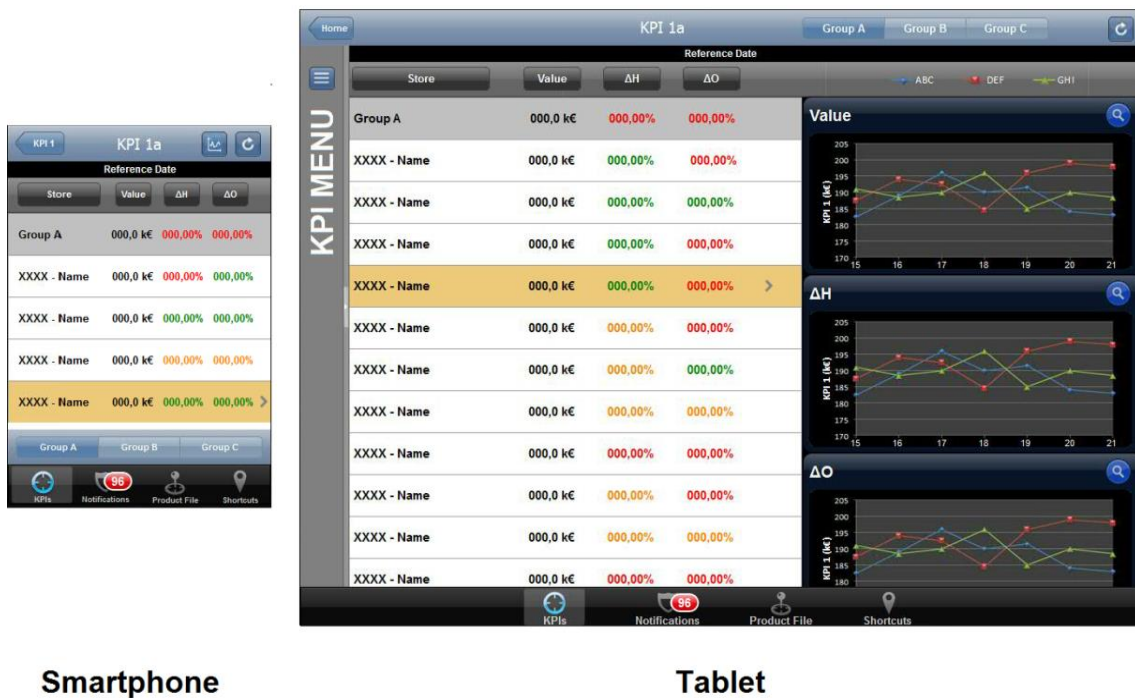
Because of its importance, notifications have larger visibility in the tablet version. The purpose of having a “Latest Notifications” section is to have a place to list chronologically all the notifications as soon as they are triggered. This helps the user to be informed, at all times, about the matters he needs to attend to. Due to screen space restriction, the smartphone version doesn’t have this section on the “Home” screen. Still, the red indicator on the “Notifications” button on the tab bar for both versions informs users of how many unread notifications there are, and pressing that button leads users to the notifications menu.

After entering in each of the KPIs sections, navigation is practically the same across all sections. In the tablet version, KPIs 1 and 2 are divided into two types (“a” and “b”) with corresponding sections for each one. In the smartphone version, due to lack of screen space, this was done with one additional screen that allows the users to select which type of KPI 1 or 2 they wish analyze (“a” or “b”), as depicted in Figure 3.11.



Figure 3.11 – Navigation between the "Home" and the "KPI 1" screens in the smartphone interface.

After selecting the desired KPI, users are then presented with the main content of the BI application. As said before, the system functional requirements were vital at this point. Taking the example presented in Table 3.6 regarding user’s need 1. All the information of that table was taken in consideration when designing the screens for KPI 1a and KPI 1b sections with every single piece of information and functionality listed in the table being included in the design of the screen. Figure 3.12 places both interfaces side by side.



Smartphone

Tablet

Figure 3.12 – “KPI 1a” screen comparison between the smartphone and tablet interfaces.

After entering the KPI 1a section of the application, users are presented with a table with four columns (Store, Value, ΔH and ΔO). The columns present the KPI 1a values for the all the stores and compare it with history (ΔH) and the budget (ΔO). The first row always refers to the group of stores selected in the segmented controls, after which all the stores are listed, with the store of the user being highlighted in orange so the user can quickly identify it among the others. Values are once again colored to give users a faster perception of which values are “good”, “normal” or “bad”.

As defined in Table 3.6, there is a “Back” button on the top left corner of the screen, a “Refresh” button on the top right corner of the screen, segmented controls to toggle between store group A, B or C, buttons on each header to sort rows and a “Chart” button to the left of the “Refresh” button to access the respective chart (only in the smartphone version because the tablet version presents the charts side by side with the data).

The differences between the smartphone and tablet KPI sections’ interfaces are mainly due, once again, to the size limitations of the smartphone screen. For example, the tablet interface places the charts side by side with the data while the smartphone interface provides a button for that purpose on the navigation bar. Other small difference resides on the placement of the segmented controls. The smartphone interface has them on a toolbar just above the tab bar and, according to the guidelines from iOS, when transitioning from iPhone to iPad, the controls placed in the toolbar should migrate to the navigation bar (Apple Inc., 2012), hence the segmented controls being placed at the top in the tablet interface.

Also, the size of the tablet interface’s buttons and text suffered an enlargement when compared with the smartphone version. The space between buttons and text also increased in order to make everything clearer and to give a more relaxed feel to the layout.

The tablet version has a “hidden” side bar on the left side of the interface, which can be quickly revealed by pressing the “Hide/Show” button above the “KPI Menu” reference. This feature is broadly used in smartphone and tablet applications. Some examples include Facebook, Google[®] and Evernote[®]. It is a simple yet useful functionality that allows users to change between KPI sections without having to return to the “Home” screen. This particular functionality was subject of several design iterations as shown in Figure 3.13.

The initial design consisted in a simple sidebar without a button to conceal it. However, a significant amount of horizontal space was taken by something that wasn’t needed all the time, shortening the space between each column.

This situation led to the creation of a “hidden” sidebar. The first iteration had the “hidden” sidebar without a “Hide/Show” button. This sidebar would be revealed by a simple “swipe” gesture which is supported by every mobile OS. The drawback of this version related to users’ possible unawareness regarding that functionality and how to trigger it. Thus, three more versions were designed with a “Hide/Show” button. These versions only differed in the placement of the said button and in the existence/inexistence of the “KPI Menu” reference.

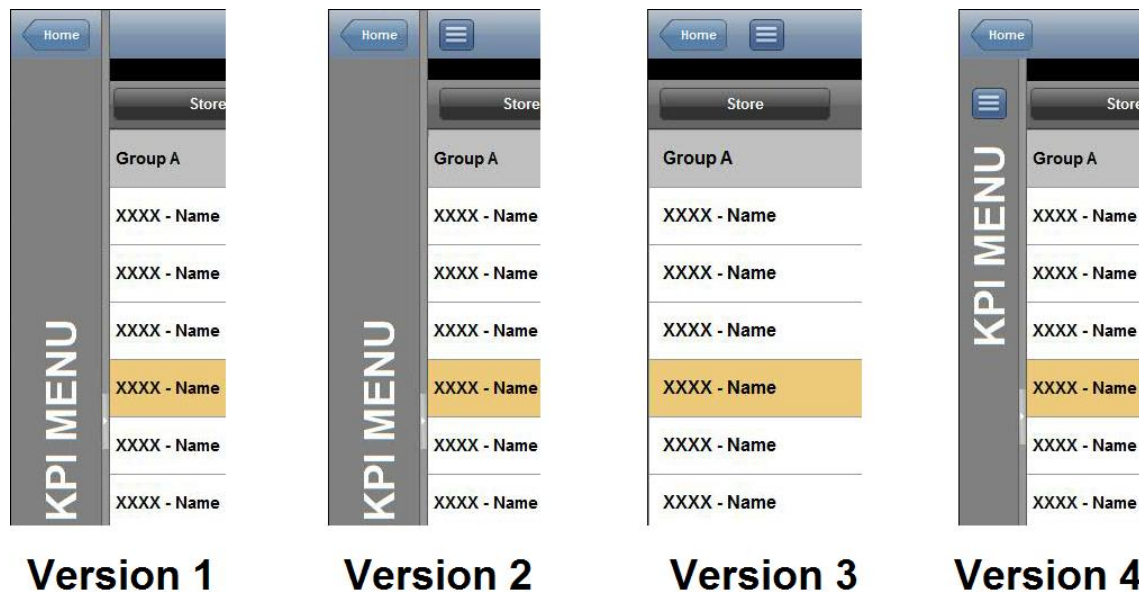


Figure 3.13 – Several sidebar design versions.

Instead of guessing which version would be more appropriate, this situation proved an excellent opportunity to involve a group of representative users and ask them which version they preferred. This procedure is commonly known as participatory design (Nielsen, 1993). Until this point the users’ participation in this phase was exclusively content-wise, correcting mismatches between what KPI details they actually needed and what was the designers’ perception of what they needed. So, for the first time, their opinion would have an impact on the actual design of the interfaces.

To obtain the maximum benefits from user involvement, it is necessary to present each suggested design in a form the users can understand (Nielsen, 1993). Therefore all four versions were presented to three representative users in the form of a small slideshow where they could interact with each version and give their opinion on which they preferred.

After interacting with every version through a simple slideshow build to simulate the sidebar functionality, users would proceed to vote on which versions they preferred (they could vote on more than one version). At the end, version 4 got three votes (maximum possible) while the

remaining three versions got only one vote each. And so, the decision was made to incorporate the sidebar from version 4 in the tablet interface. Its functionality is depicted in Figure 3.14.

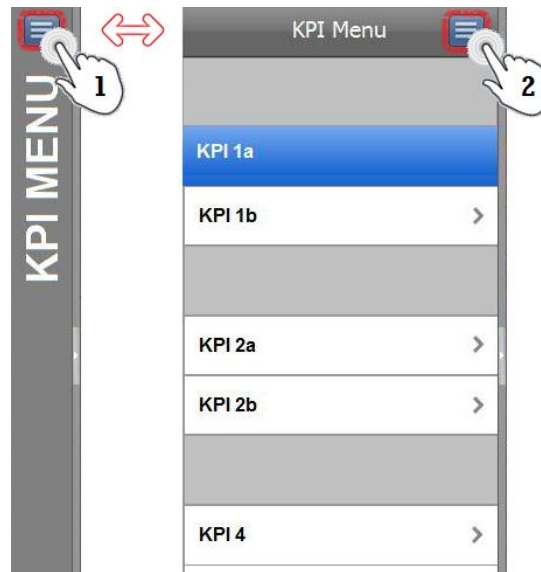


Figure 3.14 – Expanding the KPI sidebar.

Another important element present in the KPI sections are the charts. These are crucial in applications of this nature because they provide a different perspective of the KPIs performance. While the values are extremely important, the charts allow users to detect trends and patterns on the performance of the KPIs over time. In the tablet interface, all three charts are always in display, in the smartphone interface users have to press the “Chart” button to summon the charts gallery. This procedure is displayed in Figure 3.15.



Figure 3.15 – Connection between the “KPI 1a” and the “KPI 1a Chart” screens in the smartphone interface.

After entering the “KPI 1a Chart” screen, users are presented with the chart referring to the “Value” column by default. At this point users can change the chart by swiping left or right. The chart order is the same as the columns on the “KPI 1a” screen: “Value” ↔ ”ΔH” ↔ ”ΔO” ↔ ”Value”. The three dots at the bottom of the “KPI 1a Chart” screen inform users about the existence of more charts (one dot per chart) and the current position of the user is highlighted by a brighter dot. The iOS guidelines recommend that this indicator should be placed at the bottom edge of the screen where it doesn’t interfere with the user’s interactions with other elements of the interface (Apple Inc., 2012).

Users can also analyze in detail each series of the chart by pressing the blue magnifying glass button at the top right corner of the screen. By doing such, a slider appears at the bottom of the screen, replacing the three dots indicator. The slider is horizontally positioned at the end of the x-axis, selecting the most recent point by default. The respective series label as well as the selected point’s date and value are listed above the chart. These elements change depending on the point and series users are viewing. Users can then slide left or right and check other values or press the slider to change the series in analysis. After finishing analyzing a chart, users can either press the red “X” button that replaced the blue button to dismiss the slider and return to the charts gallery or press the “Back” button at the top left corner to return to the “KPI 1a” screen. An example of the navigation in a chart is shown in Figure 3.16.

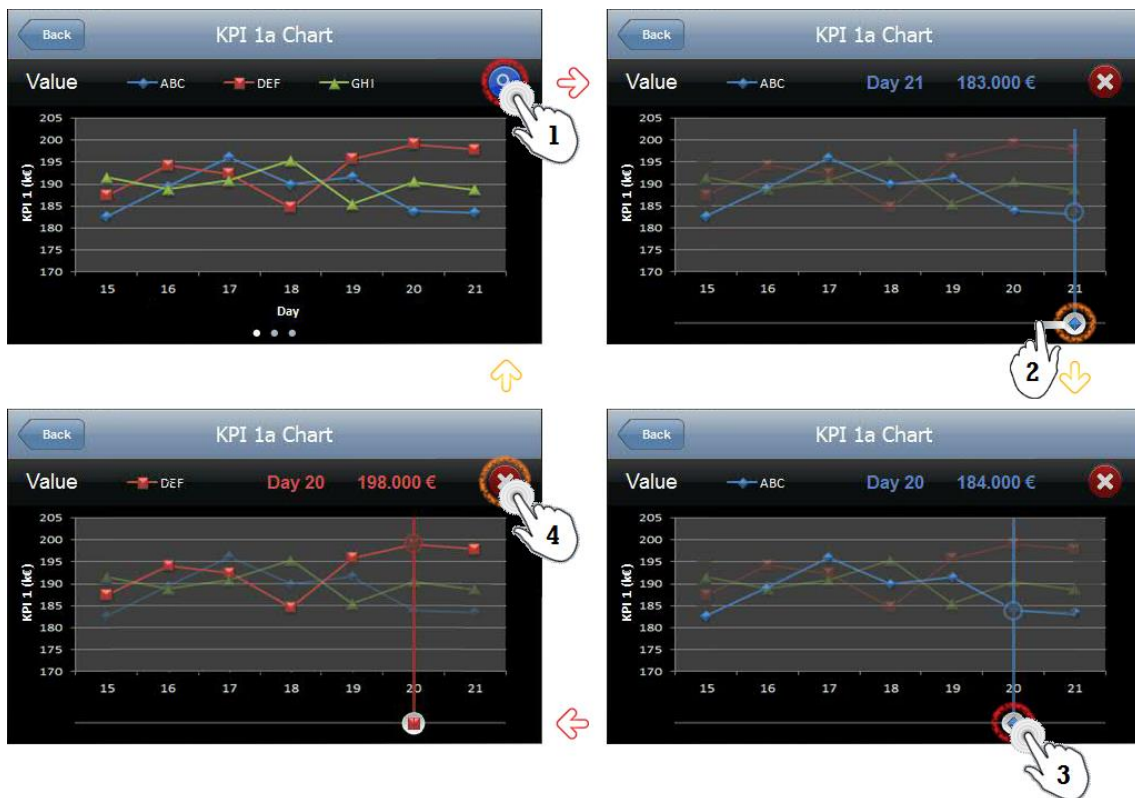


Figure 3.16 – Navigation in the “KPI 1a Chart” screen.

All the KPI sections have similar navigation and architecture, only differing in the type of values shown. Therefore, it's time to advance to the next section of content present in the BI application- Notifications.

This section is a very important part of the BI application. Rather than detecting potential problematic situations in the KPIs performance while they're analyzing them, users are notified with simple messages whenever those situations occur. Of course, not every single type of situation was contemplated in this notification section. As discussed in subchapter 3.2, one of the outcomes of the final meeting where all the users were gathered, was the addition of user's needs 5, 6, 10 and 29 to the range of notifications. These needs referred to KPIs which value would be increased if presented to users as notifications. The other types of notifications were already defined in Phase 1 while collecting the user's needs.

There are a total of eight different types of notifications. Three of them notify the user about situations occurring in real time on their workplace while the remaining five are notifications of things that happened on the day before. In both interfaces, users have awareness of the amount of notifications unopened by looking at the red sign over the "Notifications" button in the tab bar.

After entering the notifications section, users are presented with eight buttons. Each button refers to a different type of notification and also has a red signal that indicates the amount of unread notification of that specific type. Figure 3.17 depicts both versions of the "Notifications Menu" screen.



Smartphone

Tablet

Figure 3.17 – “Notifications Menu” screen comparison between the smartphone and tablet interfaces.

The tablet interface version follows the same concept of the KPI sidebar. However, this sidebar doesn't have the "Hide/Show" button because users will always want to have the perception of how many notifications exist for each type when they enter this section of the BI application.

On the tab bar, the highlighted button shifted from "KPIs" to "Notifications" to signal the change of section. The "Notifications" icon should now have a blue glow while the "KPIs" icon should be colored in grey. This change in the icon colors isn't represented in Figure 3.17 due to the limitations of the Prototyper software library which only provides a single version of the tab bar icons.

Also, there is a "Clear" button for each notification type. By pressing this button, users clear the number of unread notifications. On the smartphone version, due to the screen size limitations, this button is located on the row of each notification type while on the tablet version this button appears on the top right corner of the screen when the users selected a notification type. The functionality of the "Clear" button is shown in Figure 3.18. Only the tablet version is shown, but both versions behave in the same manner.

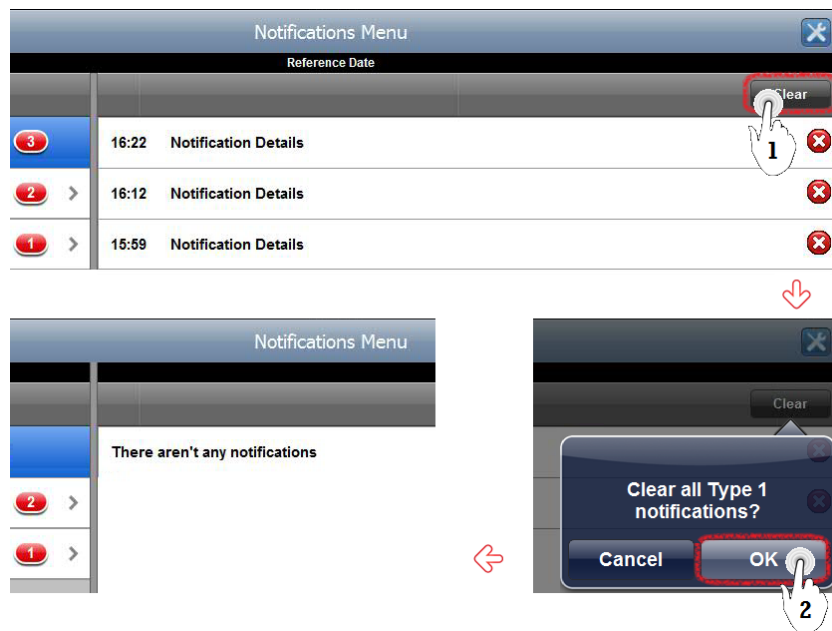


Figure 3.18 – "Clear" button functionality.

In addition to being able to clear all notifications of a certain type by pressing the "Clear" button, users can also clear notifications one by one by the pressing the red "X" button.

On the top right corner of the "Notifications Menu" screen there is the "Configurations" button where users can define the conditions that trigger each notification type or simply activate/deactivate notifications. Both versions interfaces are represented in Figure 3.19. These

configurations intend to give the application a more personalized nature, because each user can choose different parameters that best suit their interests.



Smartphone

Tablet

Figure 3.19 – Notifications “Configuration” screen comparison between the smartphone and tablet interfaces.

Both versions have exactly the same options; the only difference resides in the presentation of the information. In the smartphone version users have to expand each notification type options by pressing the blue arrow on the right while in the tablet version all the options are presented without the need of expanding/collapsing different groups of options.

There are only five groups of options because notification types 4 and 8 do not require configuration and both notification types 5 and 6 have the same options which translates into only one group of options for them.

By pressing the “Bell” button, users can activate (Green) or deactivate (Grey) each notification. Also, to select the appropriate setting for each option, users just press the yellow buttons to summon a spinner (called picker in Apple guidelines) popup and then they just have to spin it and select the desired setting. Spinners are useful for data picking and provide a quick way to select one value from a set. In the default state, a spinner shows its currently selected value (Android, 2012). This process can be seen in Figure 3.20.

After finishing the notifications parameterizations, users can save their configurations by pressing the green “Save Configurations” button, after which an alert dialog appears acknowledging that

the configurations have been saved. If users somehow forget to press that button and exit the configurations, an alert dialog appears, prompting users to save their changes. Figure 3.21 depicts this situation.

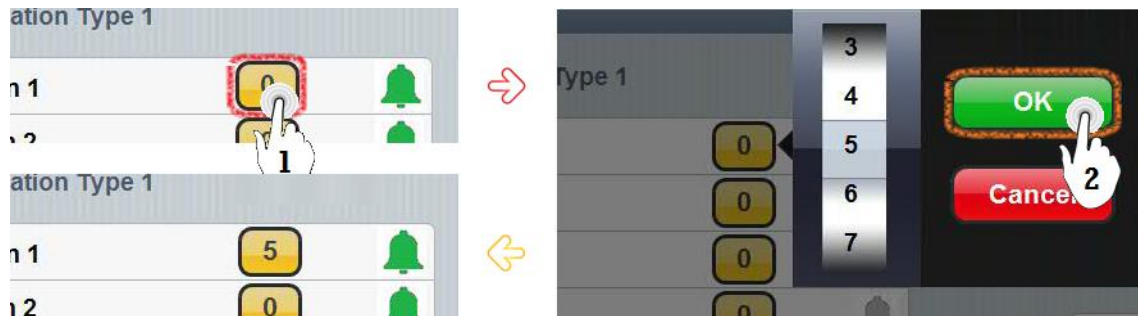


Figure 3.20 – Spinner popup to select the desired option.

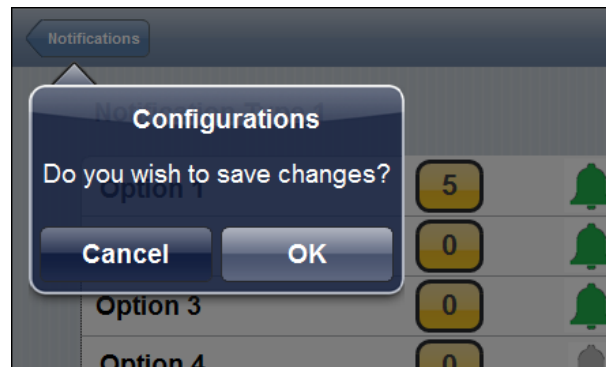


Figure 3.21 – Saving notifications configurations.

Communicating with users in such ways can help mitigate some uncertainty regarding things that have happened or will happen. Confirming or acknowledging can also prevent users from making mistakes they might regret (Android, 2012).

The following section of the BI application refers to the “Product File” which is a section where users can check many indicators of a certain product by entering the product’s code or by reading its barcode with the smartphone/tablet camera. This functionality is already present in many applications which are available for free on the applications market. Figure 3.22 places both versions of this screen side by side.

While the smartphone version only presents the options to enter the product code or read the product barcode, the tablet version also presents a blank product file. After entering the product code or reading its barcode, the blank fields in the product file are filled with the data pertaining to the product. In the smartphone version interface, the product file is accessed in a similar way. Figure 3.23 presents both ways of accessing the product file in the smartphone. Note that the

interface representing the barcode reader is taken from an application called “Barcode Scanner” developed by ZXing Team and was used to better illustrate the concept of reading a product barcode.



Smartphone

Tablet

Figure 3.22 – “Product File” screen comparison between the smartphone and tablet interfaces.



Figure 3.23 – Both ways to access the product file.

The indicators included in the product file were defined in the group meeting from Phase 2, where the users were asked which indicators they considered important to be included in the product file.

The question was asked to the group rather than individually so users could discuss with one another and reach an agreement on a final range of indicators that satisfied everyone.

The last section of the BI applications is called “Shortcuts” and refers to several functionalities that aim to satisfy some user’s needs. These functionalities are either native to the equipment (contact list, e-mail, internet browser, etc.) or available for free in applications downloadable from the OS app stores (notepad, calendar, etc.). Figure 3.24 places both interfaces side by side.



Figure 3.24 – “Shortcuts” screen comparison between the smartphone and tablet interfaces.

The purpose of this section is to give users the ability to access these functionalities from within the application without having to exit/hide it. It also increases users’ productivity because they take less time in accessing these other functionalities than they would if they had to exit/hide the BI application and open the desired functionality.

At the top right corner of the screen there’s a “+” button that enables users to add more shortcuts to this section. When users press that button they summon the list of applications installed on their equipment and can then select whichever shortcuts they desire.

The icons used in the interfaces and seen in Figure 3.24 were downloaded from Iconfinder⁷ which is an online search engine for icons, and they intend to give a realistic representation to the section.

⁷ <http://www.iconfinder.com/>

This concludes the design of each of the four sections of content included on the BI application (KPIs, Notifications, Product file and Shortcuts). One additional screen was designed, regarding the logout process. As said before, when users wish to exit the BI application, they can either press the “Logout” button located on the top left corner of the “Home” screen or simply press the “Home” button of their physical devices. While the second method is instantaneous and doesn’t alert the user, the first method requires confirmation, because users should be alerted to the consequences (exiting the application) of their actions (pressing the “Logout” button) before deciding to proceed (Android, 2012). Figure 3.25 depicts the message presented to users after they press the “Logout” button on the tablet interface. The smartphone version is similar.

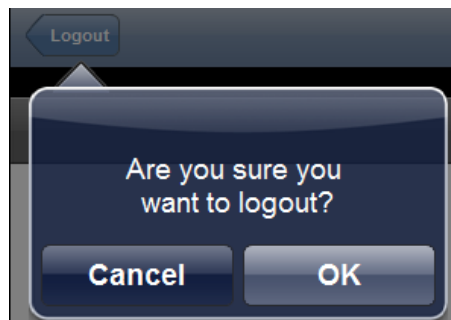


Figure 3.25 – Logout confirmation.

This concludes the analysis of the process of designing the interface prototypes for the smartphone and tablet. Of course, due to the extensive nature of the process, only a fraction of screens were shown and detailed in this dissertation. A total of 199 screens were designed, 122 for the smartphone and 77 for the tablet. Figure 3.26 illustrates these numbers.

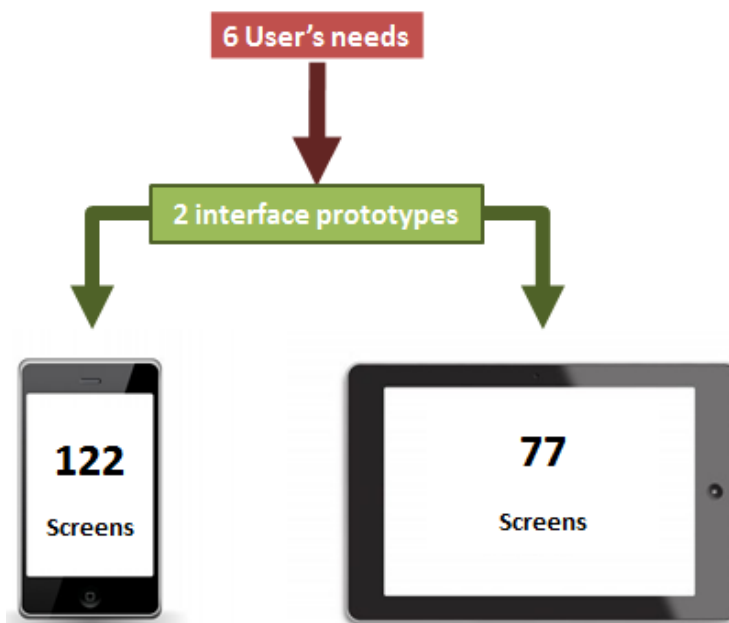


Figure 3.26 – Six user's needs originated the design of 199 screens.

Interface prototypes development

Naturally, the tablet interface prototype took considerably less screens (approximately 37% less than the smartphone interface) because more information could be presented in a single screen.

4. Usability evaluation and testing

This chapter explains how the interface prototypes were simulated in order to be evaluated and tested. The evaluation was comprised of a list of heuristics collected from various sources while the testing was performed through the Cognitive Walkthrough method applied to several participants. At the end of the chapter, the results from the testing were analyzed and discussed.

4.1. Interface simulator creation

With the conclusion of Phase 3, the following step was to build a simulation of the actual BI application running on both systems. The Prototyper software allows users to export their designs to a Portable Document Format (commonly known as PDF). After exporting both sets of screens, each screen was copied from the PDF to Microsoft Paint or other software with basic tools to editing images. Each screen (or image) was then cropped to remove unnecessary elements.

After having the 199 images in separate files, two Microsoft PowerPoint® files were created (one for each simulation) and each image was placed, separately, on an individual slide. In order to transform the slideshow into an actual simulation, 100% transparent shapes were placed on top of each interface button and connected to the destined slide so the only way to transition between slides in the presentation would be by pressing the actual buttons on the interface simulation. Figure 4.1 illustrates an example of the steps of this process in detail (note that the PowerPoint version is in Portuguese).

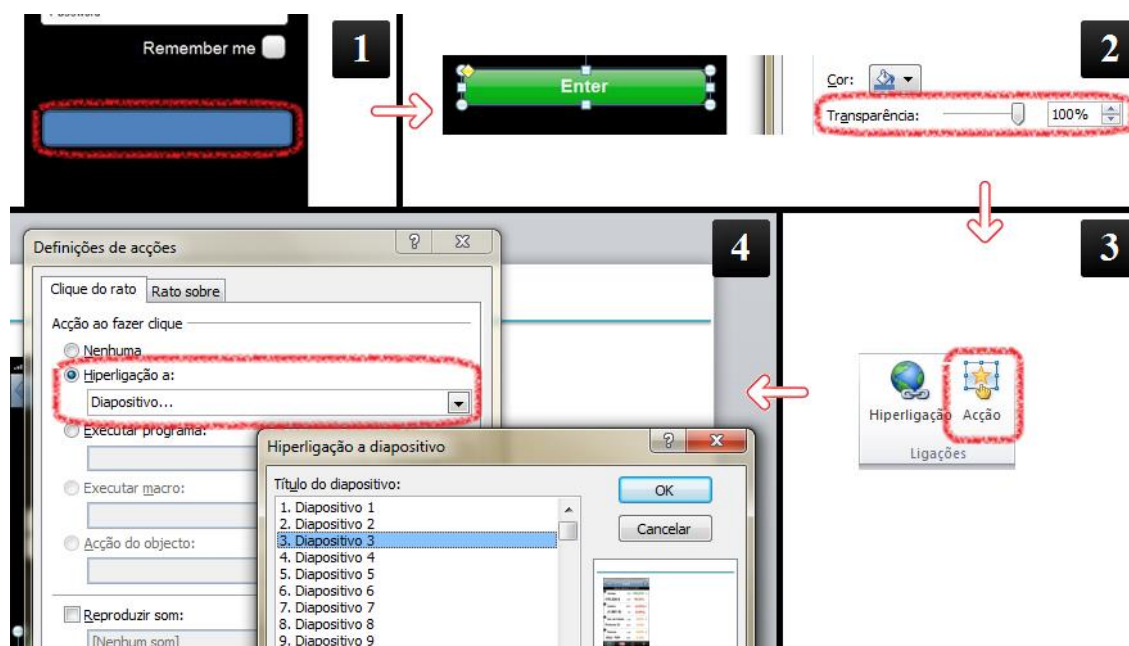


Figure 4.1 – Interface simulation using PowerPoint.

In the example shown, a shape was placed on top of the “Enter” button on the “Login screen” (1). Then, in the shape’s filling options, the transparency indicator was changed to 100% (2). In order to connect the shape to the respective slide, there is an option in the “Insert” separator called “Action” (3). After selecting this option, the next step is to select the desired slide to which the user will transit to after clicking the shape (4).

This process was repeated numerous times to establish the connections between all the screens designed.

After having both simulations prepared it was time to subject them to a heuristic evaluation in order to find potential usability problems and correct them before advancing to the Cognitive Walkthrough tests with representative users.

4.2. Heuristic evaluation

Heuristic evaluation can be performed on interface prototypes, which makes it a suitable tool for finding usability problems in the early stages of product development. Ideally, heuristic evaluation should be performed by several evaluators in order to find different usability problems (Nielsen, 1993). However, in this dissertation, only one person (the interface designer) subjected both interface prototypes to a heuristic evaluation.

These heuristics should focus on mobile applications supporting touch and multitouch interaction. Despite their use being widely spread across many devices such as self-order and information kiosks, PDAs or gaming devices (Haywood & Reynolds, 2008), the body of knowledge regarding usability guidelines for touch and multitouch interaction is still fairly limited. However there are many generic usability guidelines and rules that can be adapted to this context (Simões-Marques & Nunes, 2012).

A few studies have taken place, with the purpose of developing a list of usability heuristics to have in mind while designing mobile applications for touch and multitouch systems. These heuristics are mostly collected and adapted from widely known and used heuristics such as Nielsen’s “10 Usability Heuristics” or Shneiderman’s “Golden Rules of Interface Design”.

With this in mind a list of several heuristics and guidelines, collected from various sources, was created and both prototypes were evaluated according to them. As said before, the idea behind this procedure is to take advantage of the research done by other people and perform the best possible evaluation of the BI application interface prototypes, hoping to detect eventual usability problems before advancing to the Cognitive Walkthrough stage.

So, a group of sixteen heuristics were collected, with the aim of covering most aspects of the interfaces prototype. This group is comprised of the ten Nielsen's heuristics as well as six heuristics collected from various sources that focus specifically on the design for mobile applications.

Gong & Tarasewich (2004) propose a set of 15 practical design guidelines for mobile interfaces using Shneiderman's "Golden Rules of Interface Design" as a starting point. Some of these heuristics are very similar to Nielsen's while others like "Design for enjoyment" do not necessarily apply to a business-oriented application. Therefore, out of the 15 guidelines proposed, four were collected.

Sjöberg (2005) developed a set of 14 guidelines for touchscreen Point of Sales (PoS) interfaces. While most guidelines are specific to those type of devices and do not apply to mobile applications, there is one which can be used to evaluate the BI application in development.

The final heuristic was collected from Haywood & Reynolds "Usability Guidelines – Touchscreens" (Haywood & Reynolds, 2008) which is a list of 7 guidelines created with the purpose of helping the design and evaluation of touchscreens. While most heuristics have a physical approach such as touchscreen size and responsiveness, there is one guideline that addresses the design of icons.

Table 4.1 presents the list of heuristics collected with the purpose of evaluating both interfaces.

Table 4.1 – List of heuristics and guidelines used in the evaluation of the interface.

Heuristic		Source
1	Visibility of system status	Nielsen, 1995
2	Match between system and the real world	
3	User control and freedom	
4	Consistency and standards	
5	Error prevention	
6	Recognition rather than recall	
7	Flexibility and efficiency of use	
8	Aesthetic and minimalist design	
9	Help users recognize, diagnose and recover from errors	
10	Help and documentation	
11	Design for multiple and dynamic contexts	Gong & Tarasewich, 2004
12	Design for limited and split attention	
13	Design for speed and recovery	
14	Design for top down interaction	
15	Keep targets at reasonable size	Sjöberg, 2005
16	Icon design	Haywood & Reynolds, 2008

The evaluation method consisted in the inspection of the various elements of the several screens of both interfaces which were later compared with the list of usability principles or heuristics (Nielsen, 1993).

Hence, each heuristic will be defined in detail, after which the outcome of the evaluation for each interface prototype will be presented.

The first set of heuristics refers to Nielsen's "10 Usability Heuristics" which are widely considered the most general principles for user interface design (Nielsen, 1995):

- **Visibility of system status:** "The system should always keep users informed about what is going on, through appropriate feedback within reasonable time".

This heuristic refers mainly to the ability of the system (the BI application in this case) in keeping users informed about what it is doing and how it's interpreting their input (Nielsen, 1993). Being an application developed mainly for information retrieval, there aren't many instances where the system has to inform users of what is happening. However, there are a few situations where feedback is given to users. For instance, whenever users save their configurations, a message appears notifying them to such event as depicted in Figure 4.2.

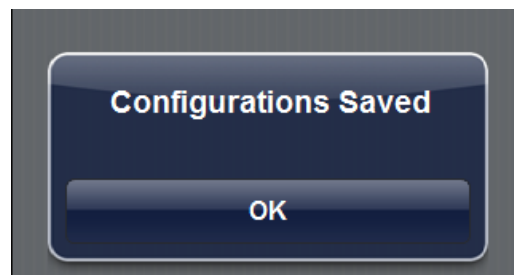


Figure 4.2 – "Configurations Saved" notification.

Also, when users wish to refresh the values and press the "Refresh" button located on the top right corner of many sections, an animation occurs with the purpose of informing users the refreshing is in process as shown in Figure 4.3.

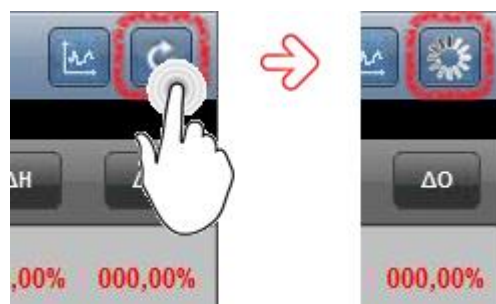


Figure 4.3 – Animation indicating that the refreshing is in process.

Also, users are always aware of their current position on the application. The grey highlighter in the tab bar at the bottom informs users about which section they're currently browsing while the title at the navigation bar informs users where they're placed in that section.

- **Match between the system and the real world:** “The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.”

Being an application intended to serve professional needs, this heuristic is extremely important, and while the terminology used in the application is not revealed in this dissertation, special care was taken to assure everything was written in a language familiar to the users, either by analyzing the terminology in the existing applications or by asking users which terms should be used.

Regarding the logical order of information, the drill down type of navigation present in the BI application is extremely intuitive and follows the architecture of the existing applications which are familiar to the users.

- **User control and freedom:** “Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo”.

Concerning this heuristic, the “Back” button present in every screen allows users to return to the previous level of the architecture. Also, the ability to “redo” isn't present, however, it isn't something mandatory for such an application because if users wish to redo their actions they just need press the button that leads to wherever they were located before. Finally, if users wish to cancel their actions (logout or save) they can simply press the “Cancel” button.

- **Consistency and standards:** “Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions”.

Platform conventions were followed by acknowledging both iOS and Android guidelines. Regarding consistency, the elements present in the application do not leave much space for dubious interpretations. Also, users' performance in the Cognitive Walkthrough later on should demonstrate that everything is completely explicit and intuitive.

- **Error prevention:** “Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action”.

Regarding error-prone situations, there were two concerns: users following a wrong path and users touching wrong buttons. Concerning the first situation, the title of the section users are currently browsing is always visible at the top of the screen (in the navigation bar). About the second situation, every button is accompanied by an icon, a label or both. Also, whenever users save or logout they're shown confirmation messages with the purpose of alerting them to that important action while giving them the opportunity to confirm or cancel it.

- **Recognition rather than recall:** “Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate”.

Regarding the minimization of users' memory load, there aren't any shortcuts or hidden options in any section of the interfaces and an attempt was made to design interfaces in such a way that navigation would be effortless for users. For instance, whenever users are presented a table of content, they can usually press the row of their store in order to drill down and advance in the information hierarchy. In order to alert users of this possibility, an arrow was placed on the row whenever such action is possible. Also, after pressing a row, the values of that row carry on to the next screen and are fixed at the top of the table so users do not have to return to the previous level of the hierarchy. Figure 4.4 illustrates both instances.

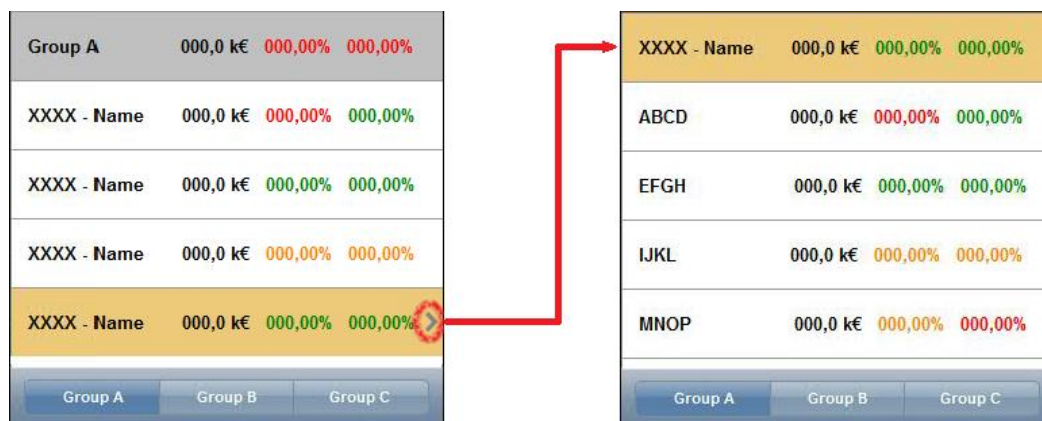


Figure 4.4 – Example of an attempt to reduce users' memory load.

The only hidden functionality present on the interfaces is the scrollbar. However this is a common practice in mobile applications. Due to the reduced available screen space mainly in smartphones, applications tend to conceal the scrollbar, only showing it when users scroll up or down. This happens in this case, but it isn't problematic because users are aware of the existence of more content than the one shown on screen (because of their experience in using the currently available applications) and the scrolling gesture comes naturally.

- **Flexibility and ease of use:** “Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions”.

Concerning this heuristic and due to the business-oriented nature of the application, there isn't anything on the BI application subject of accelerators. There are, however, some sections that provide users with a degree of customization and personalization. For instance, the “Home” screen configurations menu allows users to customize which KPIs appear on that screen. Also in the “Shortcuts” section users can add their own shortcuts to other applications they have installed in their devices.

Perhaps leisure-oriented applications or business applications that are more of the “data entry” type rather than the “information retrieval” type would need to closely follow this heuristic. In this case, users have a company policy to follow and so every user pretty much will use the application in the same way.

- **Aesthetic and minimalist design:** “Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility”.

This heuristic is very important, especially in devices with reduced screen size such as the smartphones. The entire user's needs assessment process was very important to guarantee that only important information would be present in the BI application. Having users participate in the interface design phase was also important to assure that the information was represented in the best possible way.

- **Help users recognize, diagnose and recover from errors:** “Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution”.

Again, being an application consisting almost entirely of information retrieval, there isn't much room for errors to happen, perhaps only synchronization errors. Concerning this subject, users are always presented with the date of the last synchronization.

There is however one instance where users should be shown an error message. Therefore, an additional screen was created. Whenever users wish to access a product's file, if they enter a code not listed in the database, an error message appears, alerting users to the situation and recommending them to re-enter the product code. This situation is depicted in Figure 4.5.

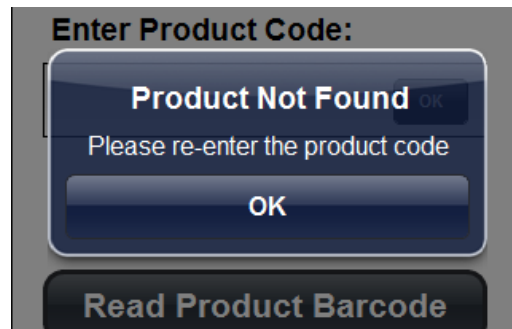


Figure 4.5 – “Product Not Found” error message.

- **Help and documentation:** “Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large”.

About help documentation, this is something that will be attended to after the BI application is finished and implemented. The company usually develops manuals for their custom applications and this will not be an exception. Since it is something that doesn't exist at this point, it can't really be evaluated. However, as said before, an effort was made to design the interfaces in a very intuitive way so users wouldn't have to recur to a manual often.

- **Design for multiple and dynamic contexts:** This heuristic refers to the variety of contexts where mobile applications can be used. Users' attention may be divided with other people, objects and situations. Environmental conditions may also affect users' interaction with the application.

In this case, users will be able to use the BI application in all of these contexts. But, since it's a business-oriented application, users will probably be more focused on its content than they would if they were using a leisure-oriented application. Regarding environmental conditions such as brightness and such, there isn't much that can be done from the designer point of view to work around this problem. The manufacturers of these types of devices should have in account these problems when developing touchscreens.

- **Design for limited and split attention:** This heuristic relates to the limited span of attention users may give the application because they may be focused on more than one task.

Regarding this subject, a study aiming to assess the extent of the fragmentation of users' attention concluded that users' have a continuous span of attention in the laboratory (or office) during 16 seconds with only one attention switch during that period (Oulasvirta, 2005). Another study

concluded that in a typical day, people spend more than two minutes on the average, on any use of electronic tool or application (González & Mark, 2004).

Again, since the BI application has a business purpose, it's reasonable to assume that users will be more focused on this particular application than they would on other applications because it's a matter of their own productivity. In the Cognitive Walkthrough tasks definition stage (see subchapter 4.3), it will be possible to know the time required for each of the representative tasks created to test the interfaces and more conclusions about this subject can be obtained at that point.

- **Design for speed and recovery:** This heuristic takes in account time constraints when interacting with mobile applications. These should provide users with a quick and secure way of saving their work so they can resume it later.

Concerning this matter, the BI application doesn't require users to save any "work" so to speak. Users can save only some configurations, and this can be done with a quick succession of two button presses (assuming the saving process isn't abnormally slow). However, every time users quit the application and return later, they are asked to input their credentials. But they can select the "Remember my credentials" box to avoid having to manually enter the username and password and accelerate the login process.

- **Design for "Top-down" interaction:** This heuristic refers to present information through multilevel or hierarchical mechanisms in order to reduce the number of interactions and potential information overload.

The BI application in development respects this heuristic in its entirety. In the "KPIs" section of the application, which is the richest section in terms of information, users can retrieve information by drilling down a hierarchical architecture, accessing only the desired sections.

- **Keep targets at reasonable size:** This heuristic suggests that smaller targets require more precision, reducing input speed. If the consequence of missing a target results solely in an erroneous action (the effect is to repeat the action), smaller targets might be applicable. Still, it's important to realize that smaller targets may affect efficiency and frequent misses increase users' frustration.

Unfortunately, this heuristic does not present concrete dimension for targets. Therefore, the dimensions were obtained from Microsoft "Touch Interaction Design" for Windows 8 (Microsoft, 2012) which recommends three types of target size as illustrated in Figure 4.6:

- ✓ 40x40 pixel recommended minimum target size;
- ✓ 50x50 pixel target size for situations where accuracy matters;

- ✓ 30x30 pixel target size for situations where touching a wrong target can be reversed with only one gesture.

Regardless of target dimensions, the minimum space between targets should be 10 pixel.

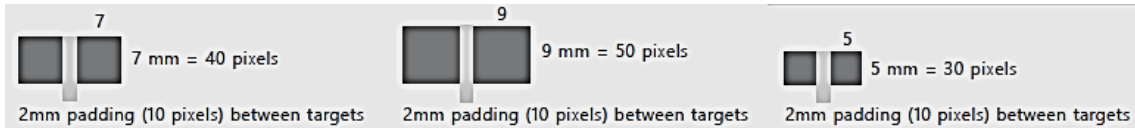


Figure 4.6 – Three different sizes for touchable targets (adapted from (Microsoft, 2012)).

Both interfaces have a wide array of different buttons spread across all the sections of the BI application. Most buttons are isolated and respect the minimum recommended size. Still, there is one particular area on the “KPIs” section that has many buttons placed together, making it an interesting subject of a closer analysis. This area is especially critical for the smartphone interface. Therefore, as seen in Figure 4.7, the targets were measured and compared to Microsoft suggested sizes in order to understand if the interface target sizes respect the guidelines. Note that the tablet interface version wasn’t addressed at this point because both target dimensions as well as the space between targets are larger in that interface version.

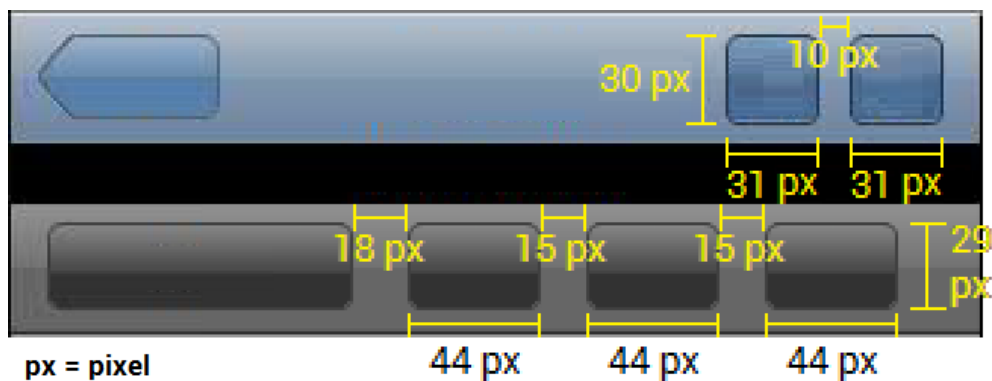


Figure 4.7 – Target and space sizes in a critical area of the smartphone interface.

Regarding the target sizes for this area of the smartphone interface, the 30x30 pixel minimum size is respected except for the height of the second row of buttons which are only 29 pixels tall. However this is compensated in their width which is 44 pixels, making them comfortable targets to interact with. The space between targets is also respected.

- **Icon design:** This heuristic suggests that icons should have a familiar design so users can relate to them. This measure intends to avoid user frustration when struggling to locate target features. Also, to avoid a confusing display, icons should be accompanied by labels whenever possible.

All the icons used in the interface prototypes of the BI application are depicted in Figure 4.8. There are five instances where icons were used: (1) in the navigation bar (chart, refresh, configurations and add buttons), (2) in the tab bar (KPIs, Notifications, Product's File and Shortcuts), (3) in the charts (magnifying glass and close buttons), (4) in the “hidden” sidebar (show/hide button) and (5) in the shortcuts in the “Shortcuts” section.

1 – Navigation bar icons

Regarding the icons present in the navigation bar, the “Refresh” and the “Add” icons are provided by Apple in its “iOS Human Interface Guidelines”.

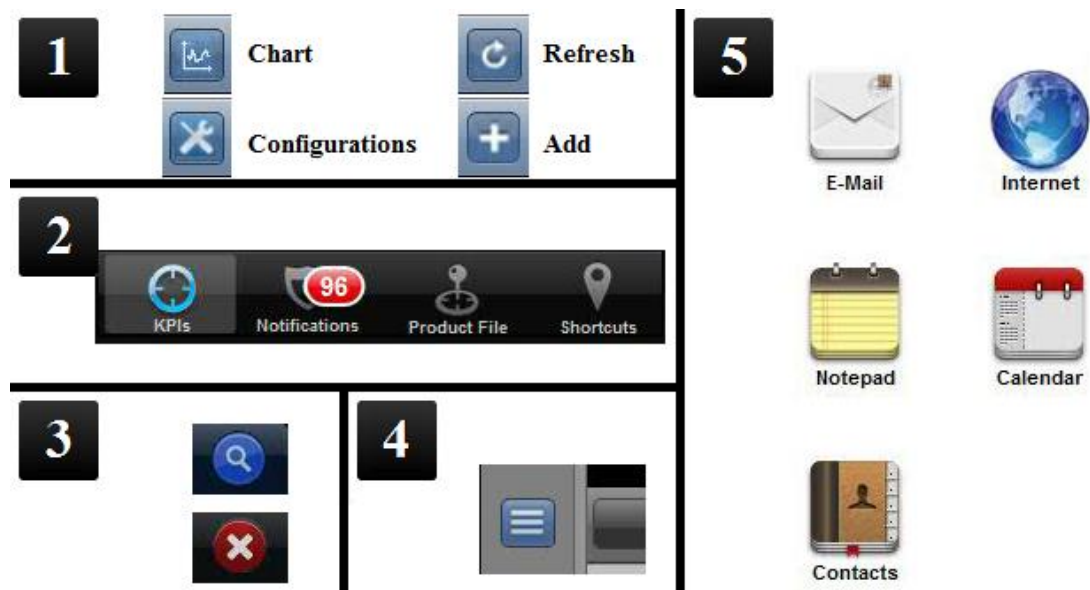


Figure 4.8 – Icons used in the interfaces.

The “Chart” icon was designed from scratch based on a quick internet search. The results obtained from the search led to the conclusion that there are mainly two ways of representing charts that users relate to: bars or lines. Since the charts present the BI application in development consist of lines, the choice fell on designing both x and y axis with a single line.

Finally, the “Configurations” icon was retrieved from Iconfinder⁸. There are two common ways of representing configurations/settings: gears or tools. The choice fell on tools but it could have been any of the two because are familiar with both.

2 – Tab bar icons

Concerning the icons from the tab bar, due to the limitations of the Prototyper software library which only provides a single version of the tab bar icons, these do not represent the respective sections. Despite being prototypes, the interfaces designed intend to represent the most of the final

⁸ <http://www.iconfinder.com/>

interfaces so, truthful tab icons should be part of the interfaces prototype. Another internet search provided very good suggestions for the tab icons as seen in Figure 4.9. It's important to point that these icons were created by IconBeast⁹ and cannot be used for free.



Figure 4.9 – New version for the tab icons.

The “KPIs” icon is actually used in the currently used BI application A with the same purpose (indicate the “KPIs” section in the tab bar). The “Notifications” icon consists in an alert which is a good representation of the section in question. The “Product’s File” icon is a barcode and the “Shortcuts” icon consists in a representation of a trifurcation because when users access the “Shortcuts” section they can then follow different paths to different applications. These icons should truthfully represent the sections they give access and since they’re accompanied by labels, users will not have a problem knowing where to navigate section-wise.

3 – Chart details icons

The next icons to be analyzed were the ones used to open and close chart’s details. The blue magnifying glass intends to tell users that they can “Zoom In” on the chart and analyze it in detail. However, the magnifying glass icon is widely used to indicate search functionalities, therefore it was slightly modified by adding a little “+” symbol inside the magnifying glass, this way users will understand that, by pressing that button, they’ll be able to access the charts’ details. This modification is depicted in Figure 4.10. Concerning the red “X” icon which closes the charts’ details, the “X” is a widely used to close windows, exit applications, delete files and more so users are already familiar with this particular icon.



Figure 4.10 – "Magnifying Glass" icon modification.

4 – “Hidden” sidebar icon

The next icon subject of analysis was the “Show/Hidden” icon from the KPI sidebar. As said in subchapter 3.4, this functionality is already used in many applications, and the way to represent it is more or less the same, a set of horizontal bars, which means that users will probably be familiar

⁹ <http://www.iconbeast.com/>

with it. Even if they're not, after finding out for the first time how to access the "hidden" sidebar, they probably will not forget it.

5 – "Shortcuts" section icons

Finally, the last set of icons used was the one present in the "Shortcuts" section of the BI application. Unlike the other icons, these icons are not meant to be present in the actual BI application, they're just representative placeholders. Android and iOS have different icons for similar applications like "E-Mail" or "Contacts" so, depending on which device users will be accessing the BI application they will encounter different icons in this section.

This ends the heuristic evaluation of the interfaces. In short, the interfaces were in compliance with 14 of the 16 heuristics, which resulted in three changes:

- The creation of one additional screen for the interfaces with the aim of alerting users when they enter an invalid product code;
- A complete overhaul of the tab bar icons;
- A slight modification in the magnifying glass icon.

After implementing these changes, the number of designed screens increased from 199 to 201, 123 for the smartphone and 78 for the tablet. The next step was to change both prototype simulations in accordance so the Cognitive Walkthrough would be performed on the simulation with the changes already implemented.

At this point, it was time to proceed to the usability testing of the prototypes by subjecting users to the Cognitive Walkthrough.

4.3. Cognitive walkthrough

The Cognitive Walkthrough is a usability inspection method intended to identify usability problems. It's a technique based on the CE+ theory of exploratory learning (Polson & Lewis, 1990). It focuses on assigning representative pre designed tasks to participants while measuring the ease with which they complete the tasks. Meanwhile participants' reactions and comments are recorded. It is a good method to apply along the early stages of product development (Simões-Marques & Nunes, 2012).

In this study, a set of six representative tasks were defined with the purpose of truthfully recreate the usage of the BI application on users' daily routines. At the same time, while defining the tasks, it was important to cover most of the different sections of the designed interfaces. Therefore there were five tasks related to different aspects of the "KPIs" section because it's the largest section of the BI application and one task referring to the "Notifications" section. However, due to

the prototypes' lack of functionality, the "Shortcuts" and "Product's File" sections were not covered in the tasks. Regarding the "Shortcuts" section, participants wouldn't be able to enter a different application, and concerning the "Product's File" section, participants wouldn't be able to enter/read a product barcode to access its file. For this reason, the tasks were defined focusing on procedures that could be realistically recreated such as analyzing certain values, charts and notifications. Also, both the "KPIs" and the "Notifications" sections will most certainly be the most viewed sections on a daily basis, with the "Product's File" and "Shortcuts" sections probably being used also on a daily basis but not so frequently. The tasks defined for testing the prototypes were the following:

- Task 1 – Analyze a certain value of the KPI 1 section;
- Task 2 – Analyze a certain value of the KPI 1a section;
- Task 3 – Analyze a certain chart of the KPI 2 section;
- Task 4 – Analyze a certain value of the KPI 4 section;
- Task 5 – Analyze a certain value of the KPI 7 section;
- Task 6 – Analyze the list of notifications of a certain type.

Each task had a similar difficulty level (all of the tasks took approximately the same number of steps and amount of time), and since the Cognitive Walkthrough was going to be performed on interface prototypes (participants weren't going to interact with any real data or access any real functionalities), in practice, the tasks consisted in users finding a particular screen.

Also, participants usually learn the product as their experience with it grows. So in order to compensate for their experience with the interfaces, the order of tasks was randomized. In other words, each participant was going to perform the tasks in a different order. This procedure is called "Counterbalancing" (Tullis & Albert, 2008).

Since there were two interface prototypes (smartphone and tablet), participants were going to be subjected to two Cognitive Walkthroughs (one for each prototype) with a two to three weeks interval between them, in order to avoid interference on the results of the second walkthrough from their previous experience with the first Cognitive Walkthrough. Specifically, participants would test the tablet interface prototype first, then, two/three weeks later they would perform the same tasks on the smartphone interface prototype. The order of tasks was not the same between prototypes either. For example, a participant would perform tasks in a certain order (e.g. T6, T2, T4, T1, T5, T3) on the tablet interface prototype and then, on the smartphone interface prototype, they would perform the tasks in a different order (e.g. T3, T6, T5, T4, T1, T2).

Regarding the participants' selection, it was based on their representativeness and availability. In other words, all the participants selected were part of the BI application's target user (company's

store managers) and all of them had the necessary time to be subjected to two Cognitive Walkthroughs in a two/three weeks period.

Concerning the size of the sample, there isn't a predefined number for the amount of participants involved. However, the chosen sample size should be based on the study goals as well as the desired error tolerance for the study (Tullis & Albert, 2008).

Since the purpose of this study is to find major usability issues in the early stages of the design and the sample is well defined and representative, the sample doesn't need to be large. However, the tolerance to error should be significant because it shortens the confidence intervals of the results and confidence intervals enable the projection of the true value of a statistic for the whole population based on what was observed in the sample. Also, according to some researchers, 80 percent of the usability issues are found within the first five participants, with the following participants finding less and less new usability issues (Nielsen, 2000). This is known as the "magic number 5" (Tullis & Albert, 2008).

Therefore, eight participants were selected. All male with an average age of 42 years (ranging between 34 and 52 years) and their working experience varies from 6 to 20 years, averaging at 13 years. It's a sample larger than the "magic number 5" which guarantees a higher probability of finding usability problems while hoping to reduce the dispersion of the results.

Regarding this study's output of the Cognitive Walkthrough, it comes in the form of a set of performance as well as self-reported usability metrics along with participants' comments.

All usability metrics must be quantifiable and represent some facet of user experience in a numeric format. There are many types of metrics that can be collected from usability tests, so it's very important to understand both the test goals as well as the users' goals in order to select the appropriate metrics to study (Tullis & Albert, 2008). As said before the study goals were to test the usability of different sections of the interface while subjecting participants to a set of tasks that replicate the actual usage of the BI application (user's goals). Table 4.2 lists ten common usability study scenarios as well as the metrics that are commonly collected in each case.

Out of the scenarios listed, this particular study intended to evaluate both the frequent use of the BI application (scenario 3) as well as its navigation and information architecture (scenario 4).

In this case it made absolute sense to collect performance metrics such as task success, task time, errors and efficiency as well as self-reported metrics. On the other hand, assessing learnability involves collecting and measuring data in multiple occasions (Tullis & Albert, 2008) which is something that unfortunately couldn't be done in this study due to the company's time restrictions. Thus, learnability was not measured. Finally, regarding card-sorting, it is a method to

organize elements of an information system in a way that makes sense to users (Tullis & Albert, 2008) it's a very time consuming process and the participants didn't had the available time required to perform this study.

Table 4.2 – Common usability study scenarios and the appropriate metrics to collect in each case (adapted from (Tullis & Albert, 2008)).

Usability Study Scenario	1 - Completing a transaction	2 - Comparing products	3 - Evaluating frequent use of the same product	4 - Evaluating navigation and/or information architecture	5 - Increasing awareness	6 - Problem discovery	7 - Maximizing usability for a critical product	8 - Creating an overall positive user experience	9 - Evaluating the impact of subtle changes	10 - Comparing alternative designs
Task Success	X	X	X	X			X			X
Task Time			X							X
Errors				X			X			
Efficiency	X	X	X	X			X			
Learnability			X							
Issues-Based Metrics	X					X				X
Self-Reported Metrics	X	X	X		X	X		X		X
Behavioral and Physiological Metrics					X			X		
Combined and Comparative Metrics		X								X
Live Website Metrics	X				X				X	
Card-Sorting Metrics				X						

After deciding which metrics were going to be collected in the study, it was time to define how to measure each.

Task success

Regarding task success, it's a mandatory metric to collect whenever a usability study involves tasks. In order to correctly measure task success, each task should be delivered to participants in a very clear way so they understand what to do and realize when they've completed it. A poorly worded task may result in participants misunderstanding it and ultimately, a failed task. The success criteria should also be defined before the testing begins. As said before, in practice, the defined tasks result in participants finding a certain screen, so that's what constitutes task success in this study.

Task success can be collected in two forms (Tullis & Albert, 2008): binary success or levels of success. Since the tasks developed for this study were pretty straightforward, meaning that participants either had success or failed them (found the desired screen or not), the choice fell on collecting task success through binary success. Each time participants performed a task, they were given a numeric score, 1 (for success) or 0 (for failure). Having task success score in a numeric format would help data treatment later on.

Task time

Concerning task time (or task completion time), it's a very important metric to be measured when the product requires users to perform tasks repeatedly. It's an excellent way to measure a product's efficiency because the time spent by participants on a task shows a lot about the product's usability. Usually, a faster task completion time translates into a great user satisfaction (Tullis & Albert, 2008).

It's important to define when to start and when to end counting the participants' time on task. In this study, it consist of the time elapsed between the start and the end of a task, expressed in seconds. Task time starts when participants press the "Enter" button on the "Login" screen and ends when participants clearly state that they've completed the task and stop interacting with the interface prototype simulation.

Errors

Measuring errors is useful to discover the possible causes of unsuccessful tasks. Errors are a useful way of evaluating the performance of participants. The number of errors made during their interaction with the interface can be very revealing, because the amount of mistakes that were

made along with the point where they were made within the interface can reflect how usable something really is.

Along with collecting the amount of errors from each participant, it's also important to define what constitutes an error. Generally an error is an action that prevents participants from completing a task in the most efficient way. There are many different types of actions that can be considered errors such as entering wrong data into a form field, making a wrong choice in a menu or list, taking an incorrect sequence of actions or failing a key action (Tullis & Albert, 2008).

In this case, two types of errors were defined: wrong choice in a menu or list (Type A) and taking a wrong set of actions (Type B). It's important to establish this difference because some errors (Type A) may be more severe in terms of usability than others (Type B). It was considered a "Type A" error every time a participant would choose a wrong option in a menu (e.g. choosing KPI 1a instead of KPI 1b) the rest of the participants' choices while they were navigating in the wrong section weren't counted as errors. On the other hand, it was considered a "Type B" error every time participants would click on a wrong button/area (for example, choosing the correct notification row, but clicking in the wrong part of the row).

Efficiency

A good way to measure efficiency is to consider the amount of effort required to complete tasks. This is typically done by counting the number of steps/actions participants take when performing each task. The more actions participants take the more effort the task requires. There are two types of effort: cognitive and physical. While physical effort involves the physical activity required to take an action (touching buttons, holding the device), cognitive effort deals with finding the correct place to perform that action (finding the correct button), deciding what action is necessary and interpreting the results of the action taken (Tullis & Albert, 2008).

When measuring efficiency, it's important to define which actions are going to be measured and count them, but only for successful tasks because counting the amount of actions taken in unsuccessful tasks will distort the test results. In this case, since users were going to test the interface prototypes in a PowerPoint simulation, each mouse click was considered an action.

Self-reported Metrics

Self-reported data is important because it can give participants' perception of the system and of their interaction with it. The best way to collect self-reported data in a usability test is with some type of rating scale such as Likert scales. Open-ended questions can also be very useful, but they are harder to analyze because they can't be converted into a numeric format. There are two moments suitable to collect self-reported data: at the end of each task (post-task ratings) and at the

end of the entire session (post-study ratings). In this study, the choice fell on collecting data on both instances. After the end of each task because it could give some insight into which tasks the participants thought were the most difficult (Tullis & Albert, 2008) and because it would be easier for participants to recall the events of the tasks if they answered these questions right after finishing them. And after the end of the entire session in order to assess participants' overall opinion of the interfaces usability.

The type of self-reported data collected after each task was the "After-Scenario Questionnaire" (ASQ) which was developed to be used immediately after a scenario completion, where a scenario is a collection of related tasks (Lewis, 1991). The ASQ consists in a set of three statements accompanied by a 7-point rating scale of "strongly disagree" to "strongly agree" and they focus on fundamental areas of usability (Tullis & Albert, 2008).

Being statements designed to be presented after an entire scenario, each refers to the tasks as a whole. Therefore, a little adaptation had to be done in order for the statements to be presented to the participants after each task rather than after the entire set of tasks. Also, the third question does not apply to the BI application because its instructions manual is yet to be developed and there isn't any "Help" functionality on the actual interface. The final statements, presented to participants were the following (note that the statements were translated to Portuguese in the questionnaire handed to the participants):

- ASQ 1 - I am satisfied with the ease of completing this task.
- ASQ 2 - I am satisfied with the amount of time it took to complete this task.

Each statement was accompanied by a 5-point rating scale of "strongly disagree" to "strongly agree". This is a modification concerning the original ASQ, but a study, assessing the impact of different Likert scales on participants' answers, concluded that both scales are comparable (Dawes, 2008). Therefore, having a 5-point or 7-point rating scale doesn't have a significant impact on participants' answers.

Concerning self-reported metrics, there was one more questionnaire delivered to participants after completing the entire set of tasks. The aim of this questionnaire was to assess participants' overall perception of usability regarding the interface prototypes. The said questionnaire is called the System Usability Scale (SUS), a reliable and low-cost usability scale that consists in ten statements (Brooke, 1996), with the odd-numbered statements worded positively and the even-numbered statements worded negatively (Lewis & Sauro, 2009). Each statement is accompanied by a 5-point rating scale of "strongly disagree" to "strongly agree". Before presenting the SUS to the participants, the word "system" was replaced by "application". The statements presented to

participants were the following (note that the statements were translated to Portuguese in the questionnaire handed to the participants):

- SUS 1 - I think that I would like to use this application frequently.
- SUS 2 - I found the application unnecessarily complex.
- SUS 3 - I thought the application was easy to use.
- SUS 4 - I think that I would need the support of a technical person to be able to use this application.
- SUS 5 - I found the various functions in this application were well integrated.
- SUS 6 - I thought there was too much inconsistency in this application.
- SUS 7 - I would imagine that most people would learn to use this application very quickly.
- SUS 8 - I found the application very cumbersome to use.
- SUS 9 - I felt very confident using the application.
- SUS 10 - I needed to learn a lot of things before I could get going with this application.

The choice of SUS was also due to its simplicity and reduced number of statements. There are other questionnaires like QUIS (Questionnaire for User Interaction Satisfaction) (Chin, Diehl, & Norman, 1988) or SUMI (Software Usability Measurement Inventory) (Kirakowski, 1994) which are more thorough but the higher number of statements may cause some discomfort on participants who may not have the available time to answer so many questions. Besides, the SUS was developed with the intention of being a tool that could quickly and easily collect a user's subjective rating of a product's usability (Bangor, Kortum, & Miller, 2008), which was exactly what was needed for this study. A study, that analyzed various SUS scores, indicates that the SUS is a highly robust and versatile tool for usability professionals (Bangor, Kortum, & Miller, 2008).

Both the ASQ as well as the SUS participants' scores were converted to numeric values in order to later treat the results in the computer. Value 1 was assigned to the "Strongly Disagree" score, while value 5 was assigned to the "Strongly Agree" score.

This concludes the presentation of the metrics collected in the study performed with the aim of testing both interface prototypes' usability.

Before describing how the Cognitive Walkthrough actually took place, it's important to note that, being an application developed for mobile equipment such as tablets and smartphones, the ideal setting would be to test both interface prototypes on actual tablets and smartphones. There are some methods capable of capturing the user experience in mobile devices like "MOD 1000" or "Mr. Tappy" which are kits developed with the purpose of capturing the user's point of view while interacting with handheld devices (Bowman, 2012). These products are shown in Figure 4.11.

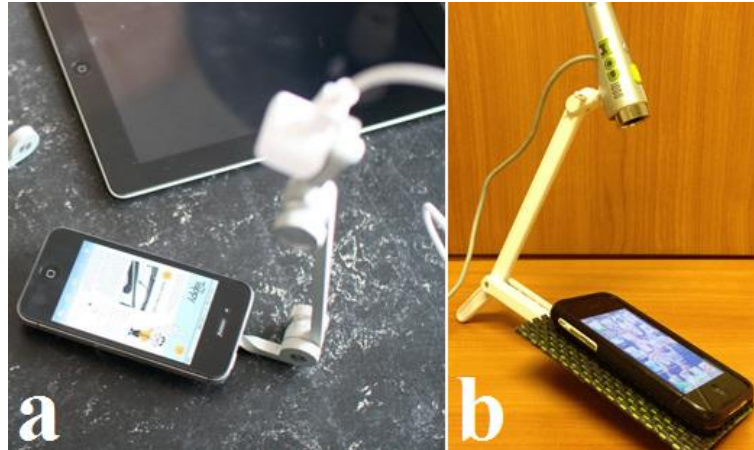


Figure 4.11 – "Mr. Tappy" (a) and "MOD 1000" (b) kits to film handheld devices (Bowman, 2012) and (Sauro, 2012).

Since this wasn't possible, the participants performed the Cognitive Walkthrough on a regular laptop, in the interface prototype simulation created. In order to count the number of actions (mouse clicks), the time of each task and record the comments of the participants while performing the tasks, the trial version of a software called "BB FlashBack" (created by Blueberry Software) was used to record the screen of the laptop where the participants would interact with the simulation.

As said before, the testing of the interface prototypes took place in two instances (first the tablet and then the smartphone prototype). The testing occurred in the participants' workplace and before starting the Cognitive Walkthrough, the participants were briefed about what was going to happen next.

The participants were told that the test was anonymous and its purpose wasn't to evaluate their performance, but the usability of the interface prototype, through a series of representative tasks. The participants were also asked to think aloud while interacting and were informed that their mouse actions as well as their comments were going to be recorded. Finally, participants were told that they were going to be presented with a small questionnaire intended to assess the usability of the interface prototype. The questionnaire consisted of eight pages. The first page was to characterize the participant's age and genre. Pages two to seven referred to each of the tasks and had the two ASQ questions along with a few lines to participants write down eventual comments about the task. The final page had the 10-statement SUS questionnaire. Participants were asked to answer according to what they really felt, and shouldn't answer positively just to please the evaluator. It was important to point out this aspect because people tend to provide more positive feedback whenever asked for self-reported data in person (Tullis & Albert, 2008).

With the briefing concluded, the participants were invited to seat in front of the computer (where the interface prototype simulation was opened and ready to use), were handed the questionnaire

(see appendix C for a copy of the complete questionnaire) and started the Cognitive Walkthrough. The participants were personally monitored while performing the tasks but their interaction with the monitor was kept to a minimum, only giving a few minor hints whenever the participants seemed lost, misread the task or forgot about its purpose. Figure 4.12 depicts a participant testing the smartphone interface prototype simulation.

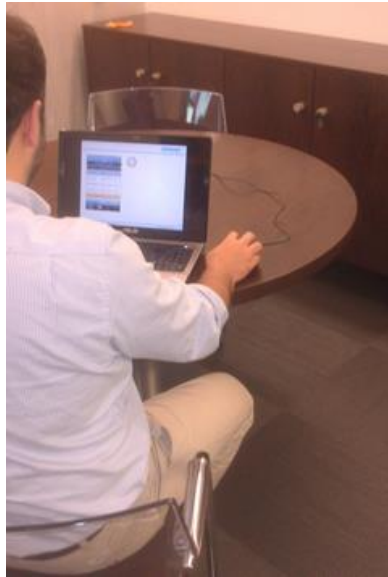


Figure 4.12 – A participant performing the Cognitive Walkthrough on the smartphone interface prototype simulation.

After finishing the Cognitive Walkthrough and completing the questionnaire, the participants were thanked for their availability and informed that in two/three weeks they would be contacted again for the second testing session (in case of having finished the first Cognitive Walkthrough).

4.4. Analysis and discussion of results

With all the recordings from both tests collected, they were viewed, one by one, and the metrics were retrieved and entered in Microsoft Excel. Table 4.3 gives an example of how the results were initially aggregated by participant. The complete set of results for all the participants can be consulted on appendix D.

The results were aggregated into three main groups: Performance Metrics, ASQ and SUS. The errors were divided by type and then summed. Also, the scores from the post-task ASQ were averaged because the first statement refers to effectiveness, the second refers to efficiency and the average refers to satisfaction (Tullis & Albert, 2008).

Having the metrics arranged by participant was a good way to initially organize them, but since the participants were not the ones subject to evaluation, there was the need to group the values in

ways that would provide significant data to evaluate the prototypes. Therefore, the results from each prototype testing were grouped by metric and task.

Table 4.3 – Testing results example (Participant 7).

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	10	0	0	0	4	5	5	5
T2	1	22	0	1	1	4	5	5	5	
T3	1	11	0	0	0	4	5	5	5	
T4	1	31	1	0	1	6	5	5	5	
T5	1	23	0	0	0	3	5	5	5	
T6	1	14	0	1	1	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	11	0	0	0	5	5	5	5
T2	1	15	0	0	0	4	5	5	5	
T3	1	16	0	1	1	7	5	5	5	
T4	1	17	1	0	1	7	5	5	5	
T5	1	17	1	0	1	6	5	5	5	
T6	1	25	1	0	1	8	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Task success

The most common way to present binary success data is by individual task, presenting the percentage of participants who were successful in each task (Tullis & Albert, 2008). This is a good starting point to understand which tasks caused more problems to participants.

In this case, all the tasks performed on both prototypes were completed successfully by all the participants. This indicates that both interface prototypes were designed in a very intuitive way. Also, the fact that the “KPIs” section was part of five tasks and its architecture was designed with many influences from applications already familiar to the participants may have contributed to having a 100% task success rate.

Despite this situation, it’s still important to know if the participants managed to successfully complete the tasks within reasonable time, efficiency and with few errors.

Task time

This metric can be displayed in various ways; perhaps the most common way is to present the average amount of time participants spent on each task. However, since time data is typically skewed, it's more appropriate to present the geometric mean. Also, to account for the potential variability across participants' times (some may take longer than normal while others may complete tasks very quickly) it's important to display confidence intervals to show the variability in the data (Tullis & Albert, 2008). Table 4.4 and Figure 4.13 show the task time results for both prototypes' testing.

Table 4.4 – Task time results for the tablet and the smartphone prototypes.

Task \ Participant	Tablet Prototype						Smartphone Prototype					
	Task time (s)						Task time (s)					
	T1	T2	T3	T4	T5	T6	T1	T2	T3	T4	T5	T6
P1	19	13	10	13	11	26	15	14	28	17	16	38
P2	16	13	38	24	13	23	13	24	26	12	24	11
P3	36	33	25	33	24	28	34	28	31	30	23	31
P4	14	38	14	17	21	13	24	30	26	24	30	30
P5	19	30	46	28	21	20	16	25	38	38	18	17
P6	27	22	25	28	28	24	29	21	35	35	21	18
P7	10	22	11	31	23	14	11	15	16	16	17	25
P8	11	28	12	17	37	20	12	17	20	27	19	25
Average (s)	19.0	24.9	22.6	23.9	22.3	21.0	19.3	21.8	27.5	23.5	21.0	24.4
Geometric Mean (s)	17.5	23.3	19.4	22.8	20.9	20.3	17.7	21.0	26.6	22.2	20.6	22.9
Confidence Interval (95%) (s)	7.3	7.6	11.3	6.2	6.8	4.5	7.2	5.0	6.1	6.9	3.8	7.3

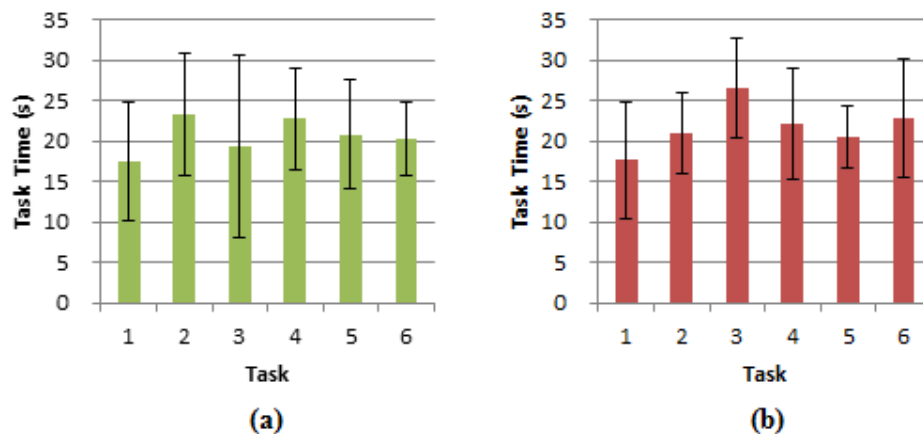


Figure 4.13 – Task time results for the tablet (a) and the smartphone (b) prototypes.

Firstly, it's important to realize that these task times do not have in account processing times that the actual BI application will eventually have. Being an application that provides users with

access to a wide range of values, which in turn are retrieved from a very large database, it's not absurd to think users will be confronted with significant processing times. However, such issue isn't of concern to this study.

Analyzing the results from both prototypes' testing, it can be concluded that participants took approximately the same amount of time successfully completing tasks on both. There is one exception, task 3, which took considerably more time completing in the smartphone prototype than in the tablet prototype (7.3 seconds more, to be exact). As said before, task 3 asks participants to analyze a certain chart from the KPI 2 section. While in the tablet interface the charts are always displayed side-by-side with the tables and thus, always visible, on the smartphone interface participants had to access the charts by pressing the "Chart" button. The existence of one more button is enough to increase participants' cognitive effort, making them think where they should press in order to access the chart.

Focusing on the tablet prototype results, it can be observed that the geometric means for all the tasks have approximate values (all the means are placed between 17.5 to 23.3 seconds interval), which leads to believe that the initial premise of defining tasks that would take approximate completion times was fulfilled. More importantly, these values mean that the various types of information spread across the different sections of the BI application are at similar distances. Users will approximately spend the same amount of time whether they're consulting values/charts on the "KPIs" section or navigating through the "Notifications" section.

Regarding results dispersion, task 3 for the tablet prototype is the task that yielded the most dispersed results between participants (has a confidence interval of 11.3 seconds). This is due to the fact that there are two different ways of completing this task. Through the KPI 1a section, it takes (ideally) 6 mouse clicks while through the KPI 1b section it takes (ideally) 4 mouse clicks. This situation, along with some participant errors (which increase the amount of time spent in completing the task), increased the dispersion of results of task 3 for the tablet prototype.

In the smartphone prototype testing results, there is the same similarity between the times of all the tasks. The geometric means for all the tasks, except task 3, have similar values (all the means are placed between 17.7 to 22.9 seconds interval).

Errors

The analysis of error data varies depending on whether the tasks have multiple error opportunities or just a single opportunity (Tullis & Albert, 2008). In this case, tasks have multiple error opportunities. Also, when analyzing errors, it's important to know how many errors participants made while trying to complete each one of the assigned tasks. Since in this case there are two

different types of errors, it's important to distinguish them graphically. Figure 4.14 presents a chart with the total number of errors performed on each task, with each type of error colored in a different way.

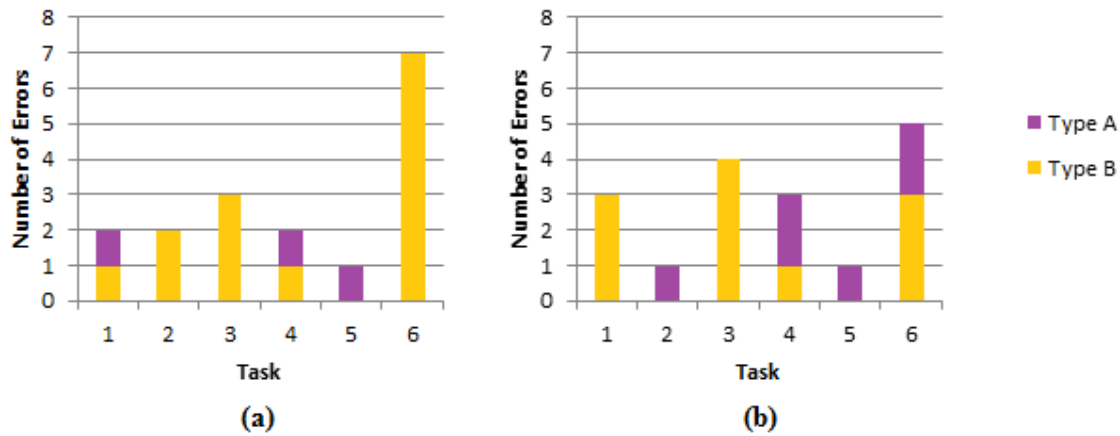


Figure 4.14 – Number of errors occurred for the tablet (a) and the smartphone (b) prototypes.

Additionally to knowing how many errors were made on each task, it's important to also know the percentage of participants that committed those errors because there may have been participants who committed more than one error while other committed none. This is illustrated in Figure 4.15.

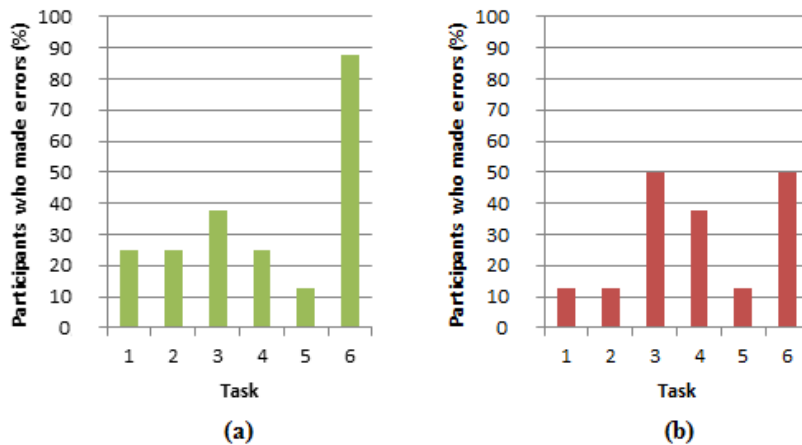


Figure 4.15 – Participants who made errors for the tablet (a) and the smartphone (b) prototypes.

Looking at the errors results, the first thing that stands out is the high number of errors happening on task 6. Specifically, seven Type B errors in the tablet prototype and five errors (two Type A and three Type B) in the smartphone prototype. This could indicate a severe problem in terms of usability in the “Notifications” section but it actually isn't. First of all, all the Type B errors for

this task happened because participants didn't click on the appropriate element within the correct section. Figure 4.16 illustrates the reason for the high number of "Type B" errors on task 6.

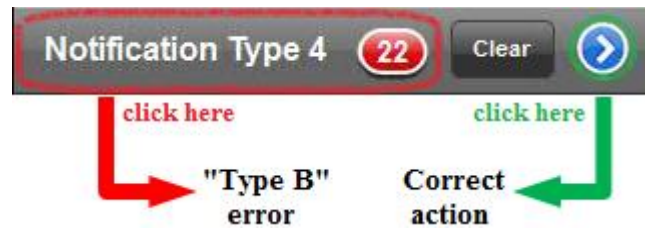


Figure 4.16 – Reason for the high number of "Type B" errors on task 6.

In order to open any list of notifications within the "Notifications" section, participants have to press the blue circular button on the right of the respective row. However, most participants assumed they could access the list by simply clicking either on the title or the red indicator. After realizing that that action didn't trigger anything, they immediately clicked on the correct button. It's safe to assume that users will quickly understand how to open the list of notifications after failing their first attempt.

The task 6 "Type A" errors that happened exclusively on the testing of the smartphone prototype happened due to participants not being initially aware of the tab bar (represented in Figure 4.17) at the bottom of the screen so they began searching for the notifications somewhere else. This situation happened only to two of the eight participants. Perhaps the said participants weren't so familiar with navigation on iOS applications because the tab bar is something very common that is present in almost every iOS application. Nevertheless, the two participants who committed the "Type A" error were asked if they would remember how to correctly access the "Notifications" section, to which they replied that after noticing the tab bar and its sections, they were not going to repeat the same mistake.

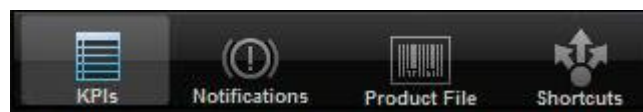


Figure 4.17 – Tab bar of the smartphone interface.

Moving on to other tasks, task 4 and 5 were a considerable source of "Type A" errors for both prototypes. These tasks ask participants to analyze certain values in the KPI 4 and KPI 7 sections, respectively. However, the reason for the considerable amount of "Type A" errors can't really be classified as a usability problem. Due to their nature, these two KPIs have some things in common, which led participants to sometimes enter KPI 4 section when the task asked was to analyze a value of the KPI 7 section and vice-versa. Once again, after realizing what were the boundaries between these two sections, participants were able to successfully complete both tasks.

Finally, task 3 also caused participants to commit a reasonable amount of “Type B” errors. Task 3 asks users to analyze a certain chart from the KPI 2 section. On the smartphone prototype the errors occur due to participants’ initial unawareness of the “Chart” button. Participants usually tried to press the respective value in order to access the chart and after realizing that didn’t work they would then search for the chart icon. In order to correct this minor problem a possible change would be to, instead of having one “Chart” button at the top that provides access to every chart, each row would have a small “Chart” icon/button to the right, informing users that they could access that specific chart by pressing that small icon/button. The initial method (present in the tested prototypes) to access the charts is illustrated in Figure 4.18.

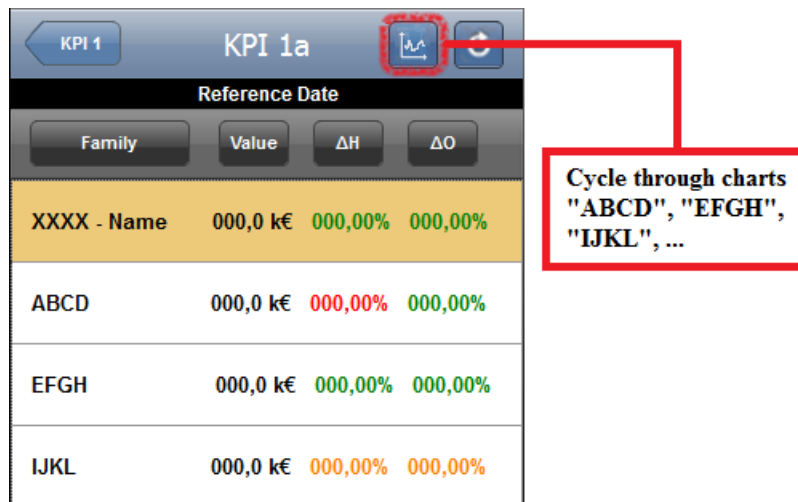


Figure 4.18 – Initial version of the method to access charts.

In the smartphone prototype subject of testing, participants had to press the “Chart” button in order to access all the charts, ordered according to the row order of the table. The proposed change is depicted in Figure 4.19.

First of all, the “Chart” button placed at the top of the screen was replaced by small individual icons placed on each row, informing users that they can access the chart. Note that these are just icons, not buttons (very small dimensions), the way to access the charts is to press anywhere on the entire row. There was the need to make small adjustments in order to create available space to the small “Chart” icons. The width of the “Family” button was slightly reduced and the spacing between the “Family” button and the “Value” button was reduced from 19 to 15 pixels. Finally, the icon itself changed from a line chart to a bar chart because in smaller dimensions bars are more noticeable than lines.

Regarding task 1 and task 2, the “Type A” errors that occurred were related to the fact of both tasks asking participants to analyze certain values from KPI 1a and KPI 1b sections which raised

some doubts on one participant. The “Type B” errors that occurred in these tasks were just minor mistakes made by some participants and aren’t relevant or require a detailed analysis.

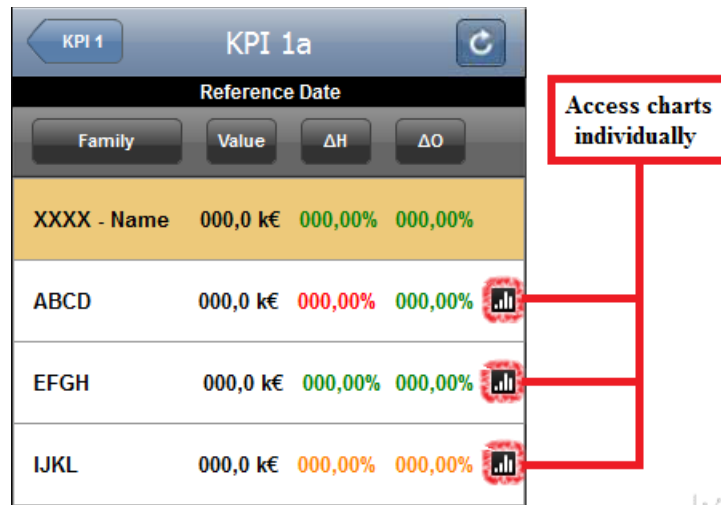


Figure 4.19 – Second version of the method to access charts.

Efficiency

As said before, in order to analyze efficiency, the number of actions (mouse clicks) taken by the participants on each task was counted. Since all the tasks were defined to take approximately the same amount of effort, by presenting the average number of actions a participant took to complete a task it’s possible to discover which tasks require the most amount of effort (Tullis & Albert, 2008). Also, it’s important to establish comparisons between the ideal number of actions required to complete each task (previously defined) and the actual participants’ performance. Figure 4.20 presents the efficiency results obtained from both prototypes’ tests.

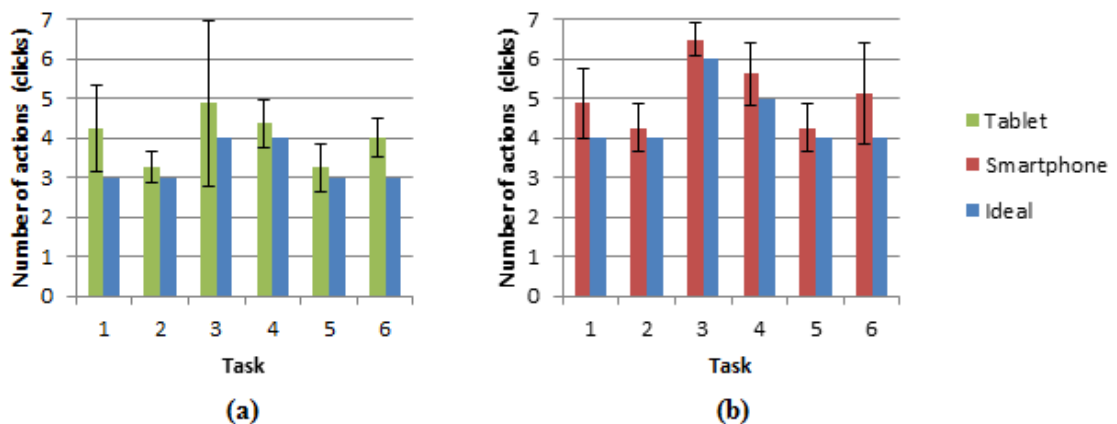


Figure 4.20 – Efficiency results for the tablet (a) and the smartphone (b) prototypes.

Looking at the results from both prototypes’ tests, it’s noticeable that the average number of actions only exceeds the ideal number of clicks by approximately 1 click for all the tasks. This

can be explained by the small number of mistakes (on average) committed by participants. The most frequent type of errors was “Type B” which usually meant the addition of one more click to the ideal number of clicks. Whenever participants would choose a wrong section (“Type A” error) they would quickly notice it and return to the right path.

Regarding the dispersion of the results, task 3 of the tablet prototype has the most (confidence interval of 2.1 clicks). This is due to some participants taking an exceedingly high number of actions (7 clicks) when compared to the majority of the remaining participants (4 clicks). Task 3 could be completed in two ways, through KPI 2a section (6 clicks) or through KPI 2b section (4 clicks). The participants who followed the KPI 2a section path took more actions and thus contributed to increasing the dispersion of the results.

This concludes the analysis of the performance metrics collected from the tests. In sum, having a 100% successful task rate in all the tasks for both prototypes was a good augury for the following results. Both the task time and efficiency metrics revealed that participants took reasonable time and number of actions to complete the tasks and the errors metric enabled to find some minor usability problems. Performance-wise it can be concluded that both interface prototypes have good usability. It’s now time to analyze the self-reported metrics.

After-Scenario Questionnaire

After collecting the participants’ answers for the two statements presented to them after each task, the results were converted to a numeric scale (from 1 to 5). It’s assumed that the distance between any two consecutive values on a Likert scale is always the same (Tullis & Albert, 2008). Averaging the results of both statements gives an overall view of the participants’ satisfaction concerning their experience with the prototypes. Figure 4.21 presents the results of the ASQ for both prototypes.

Considering the results for the satisfaction metric (average of both ASQ results) it can be concluded that the participants were extremely satisfied with the usability of both interface prototypes tasks. The lowest satisfaction rating is 4.63 for task 3 of the tablet prototype. However, it’s important to remember that, despite asking participants to answer honestly to the statements, some may have answered favorably in order to please the designer. Still, if their experience had been not so good, the performance results would have reflected it. Since that wasn’t the case, it’s safe to assume that these ratings are reliable.

After separately analyzing each of the performance and self-reported metrics’ results associated with each task, the next step involved combining most of those metrics into one single usability score for each prototype, the Single Usability Metric (SUM).

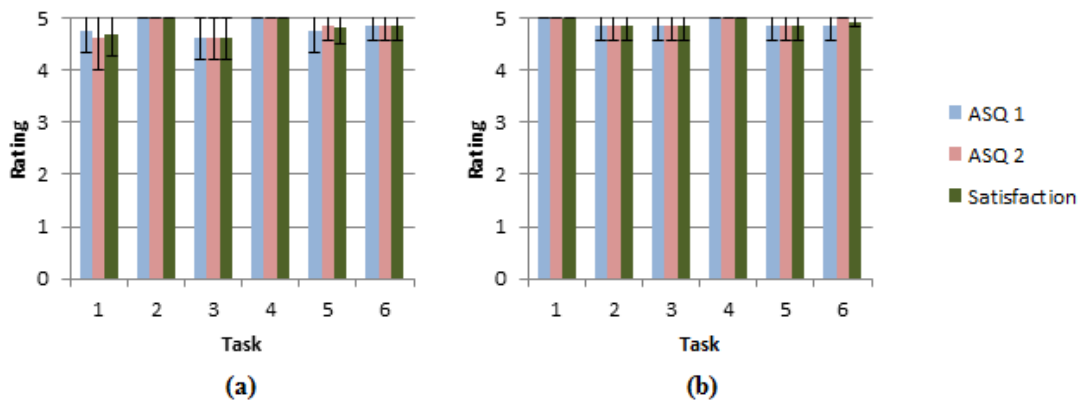


Figure 4.21 – ASQ results for the tablet (a) and the smartphone (b) prototypes.

Single Usability Metric

The SUM intends to summarize four variables (satisfaction, task time, task success and errors) into one single usability score for each task (Sauro & Kindlund, 2005). In order to calculate this score, an Excel spreadsheet created by the authors of the SUM was obtained from the following website: <http://www.measuringusability.com/SUM/>. The data entered for each participant and task was the following:

- Participant satisfaction (consisting on the average of the two ASQ answers);
- Task completion (binary success);
- Number of errors committed on that task;
- Task time (in seconds).

After entering these four values for all the tasks, it's also important to specify the confidence level and the number of error opportunities for each task. The confidence level chosen was 95% as it was the level used for all calculations. Regarding the number of error opportunities, it wasn't easy to assign a truthful value because it's somewhat subjective and hard define what constitutes an error opportunity. It was assumed that five error opportunities for each task would be a reasonable value. After entering all the data, the SUM score was calculated and presented in a bar chart as seen in Figure 4.22.

Each of the 6 tasks had a very good SUM score for both prototypes. The lowest overall score was task 6 for the tablet which scored 91.8%. This is probably due to the majority of participants incurring in a "Type B" error while attempting to complete this task. But, as it was seen before, that error was irrelevant in terms of usability. Also, an interesting pattern detected in the SUM results revealed that the smartphone prototype scored slightly better than the tablet prototype in all the tasks. This may be explained by the fact that participants tested the tablet prototype first, and

while there was a two to three weeks interval between tests, maybe it wasn't enough time for them to forget everything they learned on the first test. Still, having every task for both prototypes scoring higher than 90% is extremely satisfactory.

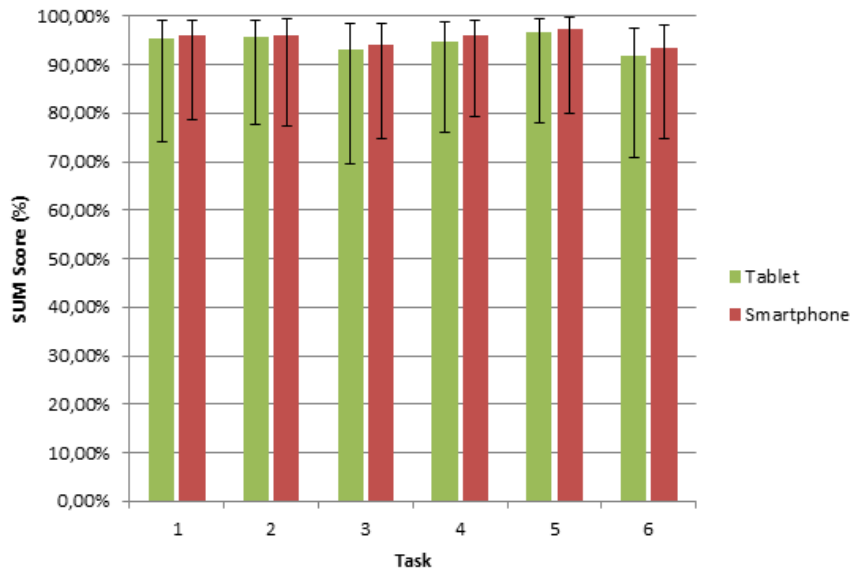


Figure 4.22 – SUM results for the tablet and the smartphone prototypes.

Finally, in order to conclude both prototypes analysis, it's very important to assess participants' perceived usability of their entire experience with both prototypes. This was done with the SUS questionnaire.

System Usability Scale

When analyzing the SUS results, it's important to understand that its purpose is to provide a single combined rating. Therefore, the individual results for each of the ten statements should not be considered (Tullis & Albert, 2008). Table 4.5 presents the SUS results for all the participants.

Both prototypes obtained very good SUS scores; the tablet prototype scored 97.81% while the smartphone prototype scored 99.06%. Having such scores in a 0-100% scale, leaves no doubt as to whether they are good or bad scores. Still, according to a study where 50 studies' average SUS scores were compiled, a SUS score higher than 80 percent can be considered pretty good (Tullis & Albert, 2008).

This ends the usability testing of the interfaces. In short, the tests resulted in very positive scores for all the measured metrics and only a small interface modification took place (the "Chart" button was replaced by individual "Chart" icons).

Table 4.5 – SUS results for the tablet and the smartphone prototypes.

Task Participant	Tablet Prototype										Smartphone Prototype												
	1	2	3	4	5	6	7	8	9	10	SUS Score (%)	1	2	3	4	5	6	7	8	9	10	SUS Score (%)	
P1	5	1	5	1	5	1	5	1	5	1	100.0	5	1	5	1	5	1	4	1	5	1	97.5	
P2	5	2	5	1	4	1	5	1	5	1	95.0	5	1	5	1	5	1	5	1	5	1	100.0	
P3	5	1	5	1	5	1	5	1	5	1	100.0	5	1	5	1	5	1	5	1	5	1	100.0	
P4	5	1	5	1	5	1	5	1	5	1	100.0	5	1	5	1	5	1	5	1	5	1	100.0	
P5	5	1	5	1	4	2	5	2	5	2	90.0	5	1	5	1	5	1	5	1	5	1	100.0	
P6	5	1	5	1	5	1	5	1	4	1	97.5	5	1	5	2	5	1	4	1	5	1	95.0	
P7	5	1	5	1	5	1	5	1	5	1	100.0	5	1	5	1	5	1	5	1	5	1	100.0	
P8	5	1	5	1	5	1	5	1	5	1	100.0	5	1	5	1	5	1	5	1	5	1	100.0	
Confidence Interval (95%)											3.05	Confidence Interval (95%)											1.56
Final Score											97.81	Final Score											99.06

Ideally, the new version of the interfaces should be subjected to the same tests or at least repeat task 3 in order to find out whether the implemented change would score better, worse or the same as the previous version.

5. Conclusions and suggestions for future work

5.1. Conclusions

The main goal of this dissertation was to follow the UCD approach in order to develop, evaluate and test two interface prototypes, one for smartphones and the other for tablets, of a mobile touchscreen BI application for store managers at a retail company. In order to successfully accomplish such task it was necessary to establish an adequate plan and schedule for all the activities that were going to take place. So, four stages were defined, each comprising several activities, with each activity being carried with the user in mind. By strictly following the schedule and planning, it was possible to reach the proposed goals and deliver two interface prototypes developed based on the user's needs collected in the first phase.

The first phase involved collecting the user's needs. At this point, even though the BI application in development was something that aimed to improve their way of working, it was extremely important the high level of commitment, dedication and availability that the company's store managers displayed. The great level of detail and objectivity given to the user's needs collected along with a thorough research on the applications that the store managers had access to, enabled a good definition of the system functional requirements. Having a representative group of users to collect needs from proved invaluable. Being an application developed with the purpose of serving professional needs, the user's need assessment shouldn't be performed in any other way.

After having all the requirements completely defined, the design of the interface prototypes for both systems (smartphone and tablet) could start. But first, it was important to understand the OS qualities and limitations, choose an adequate prototyping software and build knowledge on the OS guidelines in order to design the interfaces in a way that would meet their standards.

At this point some issues were acknowledged but were not dealt with such as different screen dimensions or having the devices screen rotation ability in account. However, it's important to remember that the purpose of this dissertation was the development of an interface prototype, not the actual interface for the end product, and so, while some things weren't addressed, it doesn't mean that the final product will not support screen rotation or different screen dimensions.

The criteria behind selecting which screens to design were the following: screens that showcase all the different functionalities and sections of the application and screens that provide testing subjects (also known as participants) with logical paths to follow in the Cognitive Walkthrough.

Having the functional requirements completely defined proved very important at this point because it worked almost as a guideline to design the screens, indicating which elements were needed for each section of content.

After designing a total of 199 screens for both prototypes, it was important to evaluate and test their usability. The evaluation consisted in collecting a set of 16 heuristics from different sources with the aim of detecting eventual flaws in the design. Since there isn't much body of knowledge regarding mobile and touchscreen heuristics, the criteria behind collecting them was to cover as many different features as possible. To this end, Nielsen's "10 Usability Heuristics" (Nielsen, 1995) were selected because they're widely used to evaluate user interface design, and while not being exclusively directed towards touchscreen applications, they fit the purpose. The remaining six heuristics collected to evaluate the prototypes were retrieved from different sources and, unlike Nielsen's heuristics, these were guidelines meant to evaluate touchscreen and mobile applications.

The heuristic evaluation revealed that the interfaces were in compliance with 14 of the 16 heuristics. Subjecting the prototypes to such evaluation was a good way to detect minor problems such as faulty icon design or the lack of an error message. With the changes from the heuristic evaluation, the total number of screens designed increased to 201. The heuristic evaluation was performed by only one person and it should, ideally, be performed by at least five usability experts because different evaluators increase the chances of finding more usability problems (Nielsen, 1993).

Finally, it was important to have the usability of the prototypes tested and subject a set of representative participants to the Cognitive Walkthrough method, which consisted in assigning them a set of six tasks intended to truthfully replicate the real usage of the BI application. However, before starting the tests, it was important to understand the goals of the users when interacting with the BI application as well as the goals of the test itself in order to define the correct set of metrics which were going to be collected. After having the metrics defined it was possible to start the tests. These took place in two different points in time (separated by a two/three weeks period) so the experience acquired from interacting with the first prototype did not influence the results from the second test. Once again, participants' commitment and availability proved crucial to the success of this stage.

At this point, one improvement could be to use more participants, even though some believe that 5 participants is enough to find the biggest errors, which is fine for testing a prototype, more participants means less dispersion in the results, which is always good to truthfully predict how the general target user will behave when interacting with the real BI application.

Still, the results from both prototypes couldn't have been better. All the metrics revealed very good results such as 100% task success for every participant, acceptable task times as well as a number of actions close to the ideal for all the tasks. The "Errors" metric was also important to detect and correct a usability problem which was causing a significant number of errors.

It is clear that having applied the User-Centered Design philosophy proved very beneficial for the end product of this dissertation (two interface prototypes with high usability that satisfy user's needs). Centering all the product development activities around users improves the chance of satisfying their needs. In this case, the initial meetings allowed understanding user's needs as well as their working environment which translated into an accurate user's needs definition. Also, being familiarized with the applications users recurred to was a very important step to accurately define the functional requirements and design the interfaces in a familiar way to users. The contribution of users in the interface design phase (participatory design) also helped in designing better interfaces. All of these factors contributed to such good results from the metrics collected as well as the ASQ, SUS and SUM scores.

Resorting to both heuristic evaluation and usability testing activities allowed a more detailed inspection of the interfaces, discovering more usability problems.

In sum, the work described in this dissertation can be considered successful. All the initial proposed objectives were successfully accomplished in the expected time and all the parts involved (the author, the company and the users) were satisfied with the results. Hopefully, the prototypes developed as a consequence of this dissertation will inspire the developers of the BI application and the design of the real interfaces will closely resemble the one proposed here.

5.2. Future work

The purpose of these prototypes was to give a very good idea to the application developers on how to create the interface of the actual final application. Therefore, more usability testing is required for the real BI application, mostly to assess the usability of its functionalities because the navigation and architecture were covered in the prototype testing.

Regarding the heuristics collected with the purpose of evaluating the interfaces, more research needs to be done on this subject because the heuristics used may not be enough to cover every aspect of mobile applications.

One thing that wasn't addressed in this dissertation was the ROI (Return on investment) analysis. It's very important to calculate the financial benefit of implementing usability improvements in a product. Also, considering the ever-changing demands in the retail industry, it's important to compare the costs of developing a non-dynamic, tailor made application that completely satisfies

Conclusions and suggestions for future work

user's needs at one point in time, but will probably need to be modified whenever users change their priorities, with using applications like MicroStrategy or PushBI™ that offer BI solutions to monitor and analyze vast amounts of data (MicroStrategy, 2007). While these solutions may not entirely satisfy every aspect of the user's needs, they are more flexible and can easily be modified whenever the company's or users' priorities shift.

References

- Airinei, D., & Homocianu, D. (2010). The Mobile Business Intelligence Challenge. *Economy Informatics, 10*, (pp. 5-12).
- Android. (2012). *Android Design*. Retrieved November 19, 2012, from Android Developers: <http://developer.android.com/design/index.html>
- Apple Inc. (2012). *iOS Human Interface Guidelines*. Retrieved November 19, 2012, from iOS Developer Library: <https://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html>
- Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An Empirical Evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction, 24*, (pp. 574-594).
- Battilana, M. (2008). *User-Centered Design and Usability: Its Role in a Project*. Retrieved March 13, 2013, from Accessites.org: <http://accessites.org/site/2008/06/user-centered-design-and-usability-its-role-in-a-project/>
- Battleson, B., Booth, A., & Weintrop, J. (2001). Usability testing of an academic library Web site: a case study. *The Journal of Academic Librarianship, 27*, (pp. 188-198).
- Birch, R. (2012). *Mobile, Handheld Devices with Touchscreens: How Perceived Usability Affects Communication*. Retrieved March 12, 2013, from Xchanges: http://infohost.nmt.edu/~xchanges/index.php?option=com_content&view=article&id=220&Itemid=250
- Björnskiöld, F., & Johansson, R. (2008). *Touchscreen GUI Design and Evaluation of an On-Device Portal*. Umeå, Sweden: Umeå University.
- Bowman, N. (2012). Retrieved February 20, 2013, from Mr. Tappy: <http://www.mrtappy.com/>
- Brooke, J. (1996). SUS - A quick and dirty usability scale. In P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), *Usability Evaluation in Industry* (pp. 189-194). Taylor & Francis.
- Capgemini. (2011). *Business Intelligence Goes Mobile: A Success Story in Business Information Delivery*. Retrieved March 12, 2013, from Capgemini: <http://www.us.capgemini.com/insights-resources/publications/business-intelligence-goes-mobile-a-success-story-in-business-information-delivery/>
- Chan, A. J., Islam, M. K., Rosewall, T., Jaffray, D. A., Easty, A. C., & Cafazzo, J. A. (2012). Applying usability heuristics to radiotherapy systems. *Radiotherapy and Oncology, 102*, (pp. 142-147).
- Chin, J. P., Diehl, V. A., & Norman, K. L. (1988). Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 213-218). Washington, D.C.: ACM Press/Addison-Wesley Publishing Co.

References

- Choi, J., & Bakken, S. (2010). Web-based education for low-literate parents in Neonatal Intensive Care Unit: Development of a website and heuristic evaluation and usability testing. *International Journal of Medical Informatics*, 79, (pp. 565-575).
- Cockton, G. (2013). Usability Evaluation. In M. Soegaard, & R. F. Dam (Eds.), *The Encyclopedia of Human-Computer Interaction*. The Interaction Design Foundation.
- Connell, I. W. (2000). *The Use of Cognitive and Other Principles in Usability Evaluation: principles versus heuristics and cumulative problem curves*. York, United Kingdom: University of York.
- Cornett, C. (2012). *Similarities and differences in iOS & Android UX design*. Retrieved November 20, 2012, from Archer: <http://www.archer-group.com/2012/mobile-category/similarities-and-differences-in-ios-android-ux-design>
- Dawes, J. (2008). Do data characteristics change according to the number of scale points used? An experiment using 5-point, 7-point and 10-point scales. *International Journal of Market Research*, 50, (pp. 61-77).
- Dinheiro Vivo. (2012). *Vendas de tablets em Portugal atingem as 222 mil unidades em 2012*. Retrieved January 18, 2013, from <http://www.dinheirovivo.pt/Economia/Artigo/CIECO067139.html?page=0>
- Dresner, H. (2011). *DAS Mobile Business Intelligence Market Study*. Dresner Advisory Services.
- Fernandes, J. M. (2012). *Vendas de smartphones em Portugal crescem 40%*. Retrieved January 18, 2013, from TeK: http://tek.sapo.pt/noticias/negocios/vendas_de_smartphones_em_portugal_crescem_40_1250784.html
- Fico, G., Fioravanti, A., Arredondo, M. T., Leuteritz, J.-P., Guillén, A., & Fernandez, D. (2011). A User Centered Design Approach for Patient Interfaces to a Diabetes IT Platform. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 1169-1172). Boston, Massachusetts: IEEE.
- Five Star Equities. (2012). *How Smartphones are now being used in major industries*. Retrieved January 5, 2013, from Yahoo! Finance: <http://finance.yahoo.com/news/number-smartphones-around-world-top-122000896.html>
- Fortulan, M. R., & Filho, E. V. (2005). Uma proposta de aplicação de Business Intelligence no chão-de-fábrica. *Gestão & Produção*, 12, (pp. 55-66).
- Furnell, S. (2007). Making security usable: Are things improving? *Computers & Security*, 26, (pp. 434-443).
- Gonçalves, D., Santos, M. Y., & Cruz, J. (2011). Implementação de um Sistema de Business Intelligence para a Análise da Qualidade de Vida Pré e Pós-Operatória. In Á. Rocha (Ed.), *Sistemas e Tecnologias de Informação na Saúde* (pp. 93-110). Porto, Portugal: UFP.
- Gong, J., & Tarasewich, P. (2004). Guidelines for handheld mobile device interface design. *Proceedings of the 2004 DSI Annual Meeting* (pp. 3751-3756). Boston, Massachusetts: CiteSeerX.

References

- González, V. M., & Mark, G. (2004). "Constant, Constant, Multi-tasking Crazyess": Managing Multiple Working Spheres. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 113-120). Viena, Austria: ACM Press.
- Haywood, A., & Reynolds, R. (2008). *Usability guidelines / Touchscreens*. Retrieved January 16, 2013, from ExperienceLab: <http://experiencelab.typepad.com/files/design-guidelines-touchscreens-1.pdf>
- Holzinger, A., Kosec, P., Schwantzer, G., Debevc, M., Hofmann-Wellenhof, R., & Frühauf, J. (2011). Design and development of a mobile computer application to reengineer workflows in the hospital and the methodology to evaluate its effectiveness. *Journal of Biomedical Informatics*, 44, (pp. 968-977).
- IDC. (2012). *IDC Raises Its Worldwide Tablet Forecast on Continued Strong Demand and Forthcoming New Product Launches*. Retrieved January 11, 2013, from IDC: http://www.idc.com/getdoc.jsp?containerId=prUS23696912#.UO_4P29riSo
- ISO 13407. (1999). *Human-centred design processes for interactive systems*. Geneva, Switzerland: International Organization for Standardization.
- ISO 9241-11. (1998). *Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability*. Geneva, Switzerland: International Organization for Standardization.
- ISO/IEC 9126-1. (2000). *Software engineering - Product quality - Part 1: Quality model*. Geneva, Switzerland: International Organization for Standardization.
- Janssen, C. (2012). *Enterprise Digital Assistant (EDA)*. Retrieved January 19, 2013, from Techopedia: <http://www.techopedia.com/definition/24234/enterprise-digital-assistant-eda>
- Kangas, E., & Kinnunen, T. (2005). Applying User-Centered Design to mobile application development. *Communications of the ACM*, 48, (pp. 55-59).
- Kavan, K. (2011). *Smartphones in the Office: The Implications for Businesses of a Smartphone in Every Pocket, Purse, and Briefcase*. Retrieved January 3, 2013, from Orange: <http://orange.eserver.org/issues/7-3/kavan.html>
- Kent, S. L. (2000). *The First Quarter: A 25-Year History of Video Games*. BWD Press.
- Kirakowski, J. (1994). *The Use of Questionnaire Methods for Usability Assessment*. Retrieved February 18, 2013, from SUMI: <http://sumi.ucc.ie/sumipapp.html>
- Kraft, C. (2012). *User Experience Innovation*. Apress.
- Lai, T.-Y. (2007). Iterative refinement of a tailored system for self-care management of depressive symptoms in people living with HIV/AIDS through heuristic evaluation and end user testing. *International Journal of Medical Informatics*, 76S, (pp. S317-S324).
- Lanter, D. P., & Essinger, R. (1991). *User-Centered Graphical User Interface Design for GIS*. Santa Barbara, California: NCGIA.

References

- Lewis, J. R. (1991). Psychometric evaluation of an after-scenario questionnaire for computer usability studies: The ASQ. *SIGCHI Bulletin*, 23, (pp. 78-81).
- Lewis, J. R., & Sauro, J. (2009). The Factor Structure of the System Usability Scale. *Proceedings of the 1st International Conference on Human Centered Design* (pp. 94-103). Heidelberg, Germany: Springer-Verlag.
- Luostarinen, R., Manner, J., Määttä, J., & Järvinen, R. (2010). User-Centered Design of graphical user interfaces. *Military Communications Conference* (pp. 50-55). San Jose, California: IEEE.
- Mano, A., & Campos, J. C. (2006). Cognitive walkthroughs in the evaluation of user interfaces for children. *Actas da 2ª Conferência Nacional em Interação Pessoa-Máquina* (pp. 195-198). Braga, Portugal: Grupo Português de Computação Gráfica.
- Mao, J.-Y., Vredenburg, K., Smith, P. W., & Carey, T. (2005). The state of User-Centered Design practice. *Communications of the ACM*, 48, (pp. 105-109).
- Meredith, W. (2008). *Best Practices of Touch Screen Interface Design*. Retrieved February 7, 2013, from Voltage Creative: <http://voltagecreative.com/articles/best-practices-of-touch-screen-interface-design/>
- Microsoft. (2012). *Touch interaction design (Windows Store apps) (Windows)*. Retrieved January 16, 2013, from Dev Center - Windows Store apps: <http://msdn.microsoft.com/en-us/library/windows/apps/hh465415.aspx>
- MicroStrategy. (2007). *Business Intelligence and Retail: Major Applications of Business Intelligence Software in the Retail Industry*. MicroStrategy.
- Moinuddin, M. (2006). *Office Business Applications for Store Operations*. Retrieved March 08, 2013, from Microsoft Developer Network: <http://msdn.microsoft.com/en-us/library/aa905316.aspx>
- Nadeem, M., & Jaffri, S. A. (2004). *Application of Business Intelligence In Banks (Pakistan)*. Karachi, Pakistan: SZABIST.
- Nagash, S. (2004). Business Intelligence. *Communications of the Association for Information Systems*, 13, (pp. 177-195).
- Navalkar, A. (2004). *Usability Engineering – Quality Approach (ISO 13407)*. Retrieved March 08, 2013, from Human Factors International: <http://www.humanfactors.com/downloads/documents/UsabilityISO.pdf>
- Nielsen, J. (1993). *Usability Engineering*. Academic Press, Inc.
- Nielsen, J. (1995). *10 Usability Heuristics*. Retrieved January 16, 2013, from Nielsen Norman Group: <http://www.nngroup.com/articles/ten-usability-heuristics/>
- Nielsen, J. (2000). *Why You Only Need to Test with 5 Users*. Retrieved March 13, 2013, from Nielsen Norman Group: <http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>

References

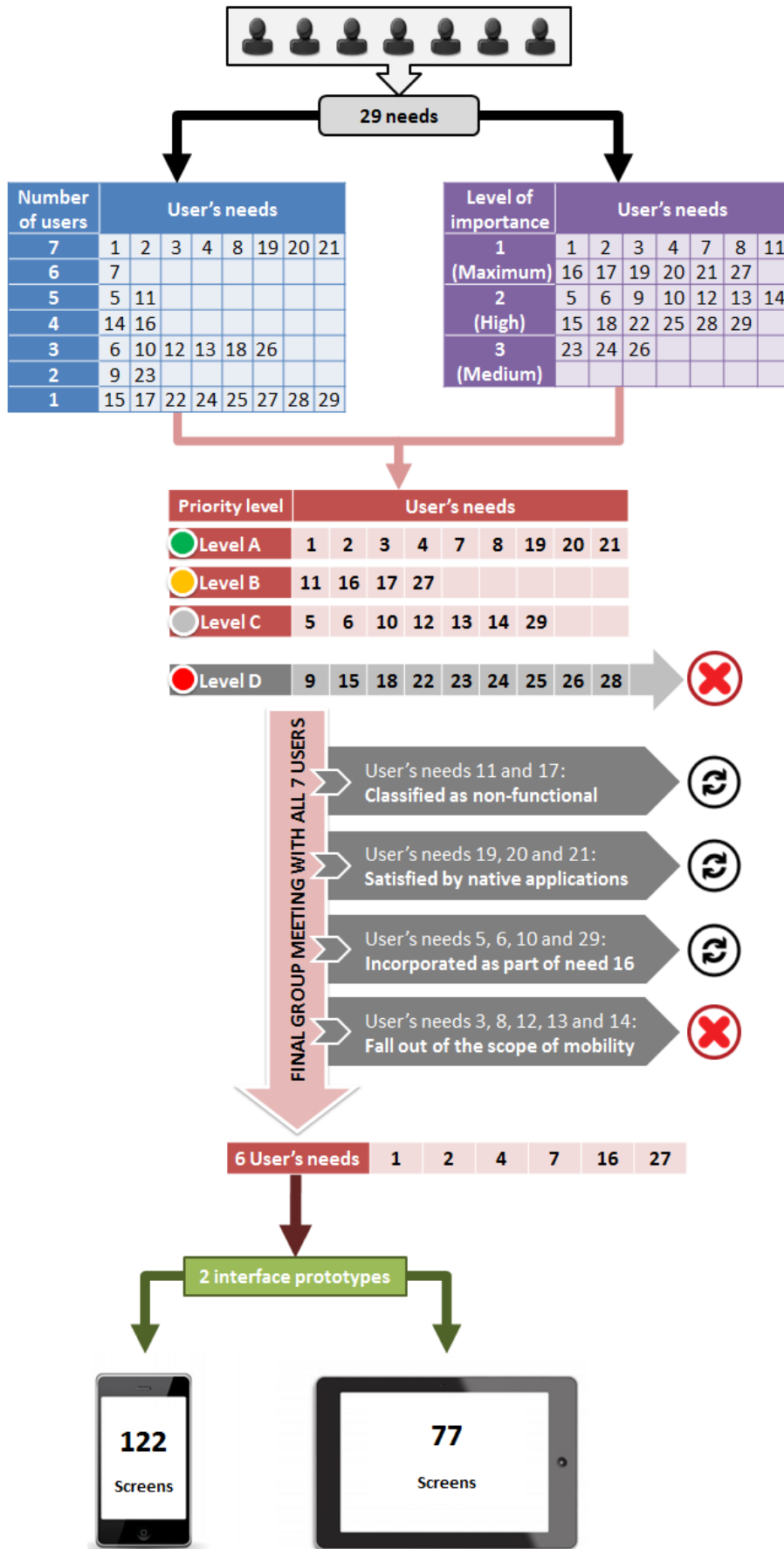
- Nielsen, J. (2012). *Usability 101: Introduction to Usability*. Retrieved January 9, 2013, from Nielsen Norman Group: <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>
- Nunes, I. L. (2006). Ergonomics and Usability – key factors in Knowledge Society. *Enterprise and Work Innovation Studies*, 2, (pp. 87-94). Monte da Caparica, Portugal: IET.
- Oulasvirta, A. (2005). The fragmentation of attention in mobile interaction, and what to do with it. *interactions*, 12, (pp. 16-18).
- Pagulayan, R. J., Keeker, K., Wixon, D., Romero, R. L., & Fuller, T. (2002). User-Centered Design in Games. In J. A. Jacko, & A. Sears (Eds.), *The Human-Computer Interaction Handbook* (pp. 883-906). Taylor & Francis.
- Park, T. (2011). *In Portrait or Landscape*. Retrieved November 28, 2012, from Thomas Park: <http://thomaspark.me/2011/10/in-portrait-or-landscape/#nav-mobile>
- Polson, P. G., & Lewis, C. H. (1990). Theory-based design for easily learned interfaces. *Human-Computer Interaction*, 5, (pp. 191-220).
- Ramakrishnan, B. (2008). *Mobile Business Intelligence for Intelligent Businesses*. Retrieved March 8, 2013, from Information Management: http://www.information-management.com/specialreports/2008_89/10001705-1.html?zkPrintable=1&nopagination=1
- Roche, M. (2012). *How Smartphones are now being used in major industries*. Retrieved January 5, 2013, from Dokisoft: <http://www.dokisoft.com/how-smartphones-are-now-being-used-in-major-industries/>
- Sampaio, P., Pereira, G., Carvalho, M. S., Telhada, J., Paisana, A., Paixão, P., & Fonseca, A. (2011). A business intelligence solution for public transportation sector. *European Concurrent Engineering Conference* (pp. 27-32). London, England: Eurosis.
- Sauro, J. (2011). *What's The Difference Between A Heuristic Evaluation And A Cognitive Walkthrough?* Retrieved March 19, 2013, from Measuring Usability: <http://www.measuringusability.com/blog/he-cw.php>
- Sauro, J. (2012). *How To Conduct A Usability Test On A Mobile Device*. Retrieved February 21, 2013, from Measuring Usability: <http://www.measuringusability.com/blog/mobile-usability-test.php>
- Sauro, J., & Kindlund, E. (2005). A Method to Standardize Usability Metrics. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 401-409). Portland, Oregon: ACM Press.
- Shneiderman, B., & Plaisant, C. (2005). *Designing the User Interface*. Pearson Education, Inc.
- Simões-Marques, M., & Nunes, I. L. (2012). Usability of Interfaces. In I. L. Nunes (Ed.), *Ergonomics - A Systems Approach* (pp. 155-170). InTech.
- Sjöberg, S. (2005). *A Touch Screen Interface for Point-Of-Sale Applications in Retail Stores*. Umeå, Sweden: Umeå University.

References

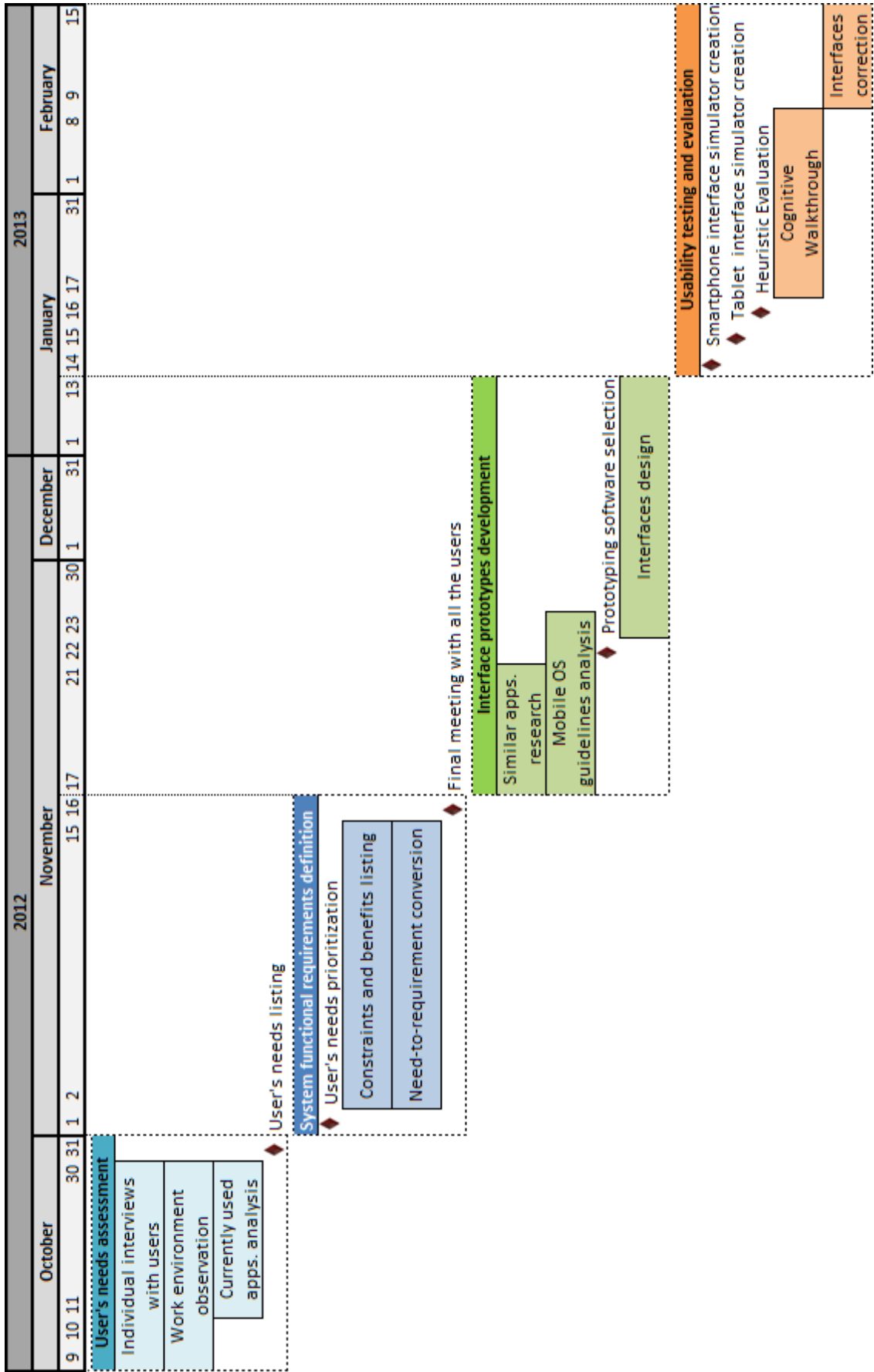
- Tabbitt, S. (2013). *Mobile Business Intelligence: Here At Last?* Retrieved March 13, 2013, from InformationWeek: <http://www.informationweek.com/software/business-intelligence/mobile-business-intelligence-here-at-las/240146333>
- Tan, W.-s., Liu, D., & Bishu, R. (2009). Web evaluation: Heuristic evaluation vs. user testing. *International Journal of Industrial Ergonomics*, 39, (pp. 621-627).
- Tedesco, D. P., & Tullis, T. (2006). A Comparison of Methods for Eliciting Post-Task Subjective Ratings in Usability Testing. *Usability Professionals Association Conference*, (pp. 12-16). Broomfield, Colorado.
- Tenreiro, P. M. (2011). *User-Centered Design of a Mobile Storytelling Application for Older Adults*. Porto, Portugal: FEUP.
- Thomsen, E. (2003). BI's Promised Land. *Intelligent Enterprise*, 6, (pp. 21-25).
- Tullis, T., & Albert, B. (2008). *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*. Morgan Kaufmann.
- Tullis, T., & Stetson, J. (2004). A Comparison of Questionnaires for Assessing Website Usability. *UPA 2004* (pp. 7-11). Minneapolis, Minnesota: UPA.
- Vai, C. T. (2010). *Business Intelligence para PME: a informação contabilística no suporte à decisão*. Lisboa, Portugal: UNL.
- Weiss, S., & Whitby, C. (2008). Usability Benchmarking: Mobile Music and Video. In T. Tullis, & B. Albert, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics* (pp. 263-271). Morgan Kaufmann.
- Zagel, C., & Bodendorf, F. (2012). User Centered Design of Innovative E-Service. *SRII Global Conference* (pp. 697-704). San Jose, California: IEEE.
- Zhang, J., Johnson, T. R., Patel, V. L., Paige, D. L., & Kubose, T. (2003). Using usability heuristics to evaluate patient safety of medical devices. *Journal of Biomedical Informatics*, 36, (pp. 23-30).
- Zhang, S., & Jeng, W. (2011). Designing a public touchscreen display system for iSchool community. *Proceedings of the 2011 iConference* (pp. 808-810). Seattle, Washington: ACM Press.

Appendices

Appendix A – Overview of the process: From the user's needs to the interface design



Appendix B - Gantt chart of the User-Centered Design activities



**Appendix C - Questionnaire handed to the
Cognitive Walkthrough participants**

Personal data

Age: _____

Genre:

M F

Figure C.1 – Page 1 (Personal data) of the questionnaire handed to the Cognitive Walkthrough participants.

Task #:

“Task description”.

1. I am satisfied with the ease of completing this task.

Strongly disagree Strongly agree

2. I am satisfied with the amount of time it took to complete this task.

Strongly disagree Strongly agree

Additional comments concerning the ease in finding the desired information:

Figure C.2 – Pages 2 to 7 (ASQ) of the questionnaire handed to the Cognitive Walkthrough participants.

1. I think that I would like to use this system frequently.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
2. I found the system unnecessarily complex.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
3. I thought the system was easy to use.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
4. I think I would need the support of a technical person to be able to use this system.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
5. I found the various functions in this system were well integrated.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
6. I thought this system was too inconsistent.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
7. I would imagine that most people would learn to use this system very quickly.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
8. I found the system very cumbersome to use.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
9. I felt very confident using the system.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree
10. I needed to learn a lot of things before I could get going with this system.
Strongly disagree <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Strongly agree

Figure C.3 – Page 8 (SUS) of the questionnaire handed to the Cognitive Walkthrough participants.

Appendix D - Cognitive Walkthrough testing results by participant

Table D.1 – Testing results of participant 1.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	19	0	1	1	6	4	4	4
T2	1	13	0	0	0	3	5	5	5	
T3	1	10	0	0	0	4	4	4	4	
T4	1	13	0	0	0	4	5	5	5	
T5	1	11	0	0	0	3	4	4	4	
T6	1	26	0	1	1	5	5	4	4	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	4	2	5	2	5	2

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	15	0	0	0	4	5	5	5
T2	1	14	0	0	0	4	4	4	4	
T3	1	28	0	1	1	7	4	4	4	
T4	1	17	0	0	0	5	5	5	5	
T5	1	16	0	0	0	4	4	4	4	
T6	1	38	0	0	0	4	4	5	4.5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Table D.2 – Testing results of participant 2.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	16	0	0	0	3	5	5	5
T2	1	13	0	0	0	3	5	5	5	
T3	1	38	0	1	1	7	4	4	4	
T4	1	24	0	0	0	4	5	5	5	
T5	1	13	0	0	0	3	5	5	5	
T6	1	23	0	1	1	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	2	5	1	4	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	13	0	0	0	4	5	5	5
T2	1	24	0	0	0	4	5	5	5	
T3	1	26	0	0	0	6	5	5	5	
T4	1	12	1	0	0	5	5	5	5	
T5	1	24	1	0	0	4	5	5	5	
T6	1	11	1	0	0	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Table D.3 – Testing results of participant 3.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	36	0	0	0	5	5	5	5
T2	1	33	0	0	0	3	5	5	5	
T3	1	25	0	0	0	4	5	5	5	
T4	1	33	0	0	0	4	5	5	5	
T5	1	24	0	0	0	3	5	5	5	
T6	1	28	0	0	0	3	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	34	0	0	0	6	5	5	5
T2	1	28	0	0	0	4	5	5	5	
T3	1	31	0	0	0	6	5	5	5	
T4	1	30	0	0	0	5	5	5	5	
T5	1	23	0	0	0	4	5	5	5	
T6	1	31	0	0	0	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Table D.4 – Testing results of participant 4.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	14	0	0	0	3	5	5	5
T2	1	38	0	0	0	3	5	5	5	
T3	1	14	0	0	0	4	5	5	5	
T4	1	17	0	0	0	4	5	5	5	
T5	1	21	0	0	0	3	5	5	5	
T6	1	13	0	1	1	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	24	0	0	0	4	5	5	5
T2	1	30	0	0	0	4	5	5	5	
T3	1	26	0	0	0	6	5	5	5	
T4	1	24	0	0	0	5	5	5	5	
T5	1	30	0	0	0	4	5	5	5	
T6	1	30	0	0	0	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Table D.5 – Testing results of participant 5.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	19	0	0	0	3	5	5	5
T2	1	30	0	1	1	4	5	5	5	
T3	1	46	0	1	1	7	4	4	4	
T4	1	28	0	1	1	5	5	5	5	
T5	1	21	0	0	0	3	5	5	5	
T6	1	20	0	1	1	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	16	0	0	0	5	5	5	5
T2	1	25	1	0	1	4	5	5	5	
T3	1	38	0	1	1	7	5	5	5	
T4	1	38	0	0	0	7	5	5	5	
T5	1	18	0	0	0	6	5	5	5	
T6	1	17	0	1	1	8	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	4	1	5	1

Table D.6 – Testing results of participant 6.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	27	1	0	1	6	4	5	4.5
T2	1	22	0	0	0	3	5	5	5	
T3	1	25	0	0	0	4	5	5	5	
T4	1	28	0	0	0	4	5	5	5	
T5	1	28	0	0	0	3	4	5	4.5	
T6	1	24	0	1	1	4	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	5	1	4	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ		
	Task			Type A	Type B	Total		1	2	Average
	T1	1	29	0	3	3	7	5	5	5
T2	1	21	0	0	0	4	5	5	5	
T3	1	35	0	0	0	6	5	5	5	
T4	1	23	0	1	1	6	5	5	5	
T5	1	21	0	0	0	4	5	5	5	
T6	1	18	0	1	1	5	5	5	5	
SUS										
Statement	1	2	3	4	5	6	7	8	9	10
Score	5	1	5	1	5	1	4	1	5	1

Table D.7 – Testing results of participant 7.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ			
	Task			Type A	Type B	Total		1	2	Average	
	T1		1	10	0	0	0	4	5	5	5
T2		1	22	0	1	1	4	5	5	5	
T3		1	11	0	0	0	4	5	5	5	
T4		1	31	1	0	1	6	5	5	5	
T5		1	23	0	0	0	3	5	5	5	
T6		1	14	0	1	1	4	5	5	5	
SUS											
Statement		1	2	3	4	5	6	7	8	9	10
Score		5	1	5	1	5	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ			
	Task			Type A	Type B	Total		1	2	Average	
	T1		1	11	0	0	0	5	5	5	5
T2		1	15	0	0	0	4	5	5	5	
T3		1	16	0	1	1	7	5	5	5	
T4		1	17	1	0	1	7	5	5	5	
T5		1	17	1	0	1	6	5	5	5	
T6		1	25	1	0	1	8	5	5	5	
SUS											
Statement		1	2	3	4	5	6	7	8	9	10
Score		5	1	5	1	5	1	5	1	5	1

Table D.8 – Testing results of participant 8.

Tablet Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ			
	Task			Type A	Type B	Total		1	2	Average	
	T1		1	11	0	0	0	4	5	5	5
T2		1	28	0	0	0	3	5	5	5	
T3		1	12	0	1	1	5	5	5	5	
T4		1	17	0	0	0	4	5	5	5	
T5		1	37	1	0	1	5	5	5	5	
T6		1	20	0	1	1	4	5	5	5	
SUS											
Statement		1	2	3	4	5	6	7	8	9	10
Score		5	1	5	1	5	1	5	1	5	1

Smartphone Prototype	Metric	Task Success	Task Time (s)	Number of Errors			Efficiency (clicks)	ASQ			
	Task			Type A	Type B	Total		1	2	Average	
	T1		1	12	0	0	0	5	5	5	5
T2		1	17	0	0	0	4	5	5	5	
T3		1	20	0	1	1	7	5	5	5	
T4		1	27	1	0	1	7	5	5	5	
T5		1	19	0	0	0	4	5	5	5	
T6		1	25	1	1	2	7	5	5	5	
SUS											
Statement		1	2	3	4	5	6	7	8	9	10
Score		5	1	5	1	5	1	5	1	5	1