Determining the User Experience Level of Operating Computer Systems in the Central Bank of Mexico

by

Jonathan Laguna Flores

B.S. in Computer Systems Engineering

Instituto Politécnico Nacional
Mexico City, Mexico, 2012

Submitted to the System Design and Management Program
in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

at the

Massachusetts Institute of Technology

June 2019

Signature of author: _____
System Design & Management Program
May 23th, 2019

Certified by: _____
Eric Rebentisch
Research Associate, Sociotechnical Systems Research Center

Accepted by: _____
Joan Rubin
Executive Director, System Design & Management Program

This page intentionally left blank.

Determining the User Experience Level of Operating Computer Systems in the Central Bank of Mexico

by

Jonathan Laguna Flores

Submitted to the System Design and Management Program
on May 23, 2019 in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Engineering and Management.

Abstract

The purpose of this thesis is to measure the user experience (UX) level of operating computer systems built in the Central Bank of Mexico. The user experience is subjective and depends on a user's interaction with the ergonomic and hedonic qualities of a product, service, or system.

A user experience framework based on the literature review was proposed. This framework was used to decompose the user experience into ergonomic, hedonic, and appeal attributes, which were used to measure the UX level of the Bank's operating computer systems. Two surveys, one for systems' users and one for systems' developers were applied in the Bank in order to collect their opinions regarding the different UX attributes (variables) of systems on a seven-point scale that made use of semantic differential (polar adjectives) technique.

The survey results were analyzed in order to identify UX opportunity areas by category of systems, as well as by UX variable. Differences among the opinions between users and developers regarding the UX level of systems were found. A strongly positive correlation between the UX level (UX index) determined through the ergonomic and hedonic variables, and the appeal (Appeal index), was found. Finally, the spotted UX opportunity areas are discussed.

Thesis supervisor: Eric Rebentisch
Title: Research Associate, Sociotechnical Systems Research Center

Acknowledgements

I want to thank Eric Rebentisch, my thesis supervisor, for giving me the opportunity to develop my ideas into this work. I dedicate this thesis to my husband, Jorge; my parents Victoria and Juan Manuel; my family; my endearing friends Selenne, Soledad, Ramón, Esperanza, Libertad, Rodrigo, Jesús, and Diego; and those whose support and company have made this a unique journey:

Carla Pérez	Antonio Moya	Eunjin Koo
Andrea Moreno	José R. I. Garza	Paulo F. López
Patricia Solis	Jessica Quezada	Natalia Coachman
Sebastián Pérez	Azka Shoaib	Karthikeyan Kuppu
Vanessa Toro	Mercedes Bidart	Viviana Siless
Gonzalo Ortega	Juan C. Constain	Alberto Barradas
Jessica Ramírez	Jazmín A. Juárez	Fabiola Ocampo
Argelia Vega	José A. Hernández	Rodolfo Méndez
Erik González	Viridiana Cabrera	David Macías
Roberto Reyes	Rodrígo Cano	Heriberto Rodríguez
Jonathan Martínez	Laurie L. Scott	Irina Singh
Mao Hattori	Ke Chia Huang	Yijia Wang

Lastly, I am also grateful for the financial support of
the Central Bank of Mexico (Banco de México).

Sincerely,
Jonathan Laguna Flores

Determining the User Experience Level of Operating Computer Systems in the Central Bank of Mexico

by

Jonathan Laguna Flores

Table of Contents

List of figures

List of tables

Chapter 1. Motivation

I am a computer systems engineer and I have worked as a software developer for the Central Bank of Mexico (*Banco de México* or just *the Bank* from now on) for almost ten years. The Bank is one of the most prestigious institutions in Mexico and takes care of the monetary policy of the country, the stability of prices, and the sound development of payment systems. Since I joined the Bank, I have worked for the Operating Systems Development Division (OSDD) as a Systems Development Analyst. This division is now part of the General Directorate of Central Bank Operations and Payment Systems (DGOSP for its acronym in Spanish).

Due to the high sensitivity of the information that it handles, the Bank has opted for in-house systems development. A highly qualified group of engineers design, build, and test either new systems or new features that existing systems require as they evolve. As engineers, we have put great emphasis on implementing reliable systems that fulfill users' requirements. We have substantial achievements when it comes to interpreting requirements and delivering robust and dependable systems, and yet I still feel like we are missing something, a spark that would allow our systems to stand out even further.

When I talked to my manager regarding how our users experienced our systems, he said: "I believe our systems are robust and that they certainly help the Bank accomplish its goals. However, I have this feeling that they are not pretty." But, why should computer systems be pretty in the first place? Is it not enough that they do what they were made for? Apparently not.

At this point I would like to introduce the term User Experience (UX), which encompasses all the experiences humans have when interacting with products, services, or systems via a user interface (e.g., keyboard, touchscreen, voice menu). UX experts believe that systems should account for both *do-goals* (i.e., functionality and usability) and *be-goals* (e.g., innovativeness, originality, aesthetics) [1]. While do-goals focus on how users interact with the system to accomplish tasks (e.g., efficiency, effectiveness, security[1]), be-goals focus more on the emotional aspects of it (e.g.,

[1] By "security" I don't mean less hackable. It's more about how secure graphical user interfaces are to use. Some interfaces' design is prone to error (or "insecure"): for example, a tool bar where the delete button

aesthetics, innovativeness, creativeness). Be-goals could be regarded as features that if present could make users feel more pleased and improve their overall evaluation of systems. Depending on how good the trade-off between be-goals and do-goals is made for a specific system, this will deliver a certain UX level (good, bad, exciting, boring, etc.).

Going back to the question about systems having to be pretty or not, one could conclude that aesthetics as part of the be-goals is an important component of the overall UX. Hence, systems' design should account for it. One could argue that aesthetics would not be as important as usability, and that might be true. Depending on the system's context, some attributes will be more important than others. For example, video game systems tend to focus more on originality, excitement, and aesthetics than they do on security. One more example could be statistical analysis software, which might focus more on aspects such as functionality and clarity than on excitement or originality.

While the software industry has been concerned about usability or do-goals, there is still much to do on the emotional side of the equation. Humans are not just about doing but also about being, and in order to come up with systems that better satisfy their needs and aspirations, engineers should start accounting for be-goals. If they deliver systems that not only meet functional expectations but that are also original, innovative, and maybe even aesthetic, chances are that users will be more satisfied when interacting with such products.

Currently, engineers at the Bank have barely been in touch with the concept of usability and there is no UX awareness regarding do-goals or be-goals. My desire to do research about UX in order to improve in-house systems has motivated me to write this thesis. I hope that the research that follows will shed some light on what the OSDD systems' opportunity areas are.

I believe that the Bank should start including UX in all its processes (i.e., systems development, marketing, payroll processes, public relations, internal audits, etc.).

---

and view button are located next to each other. Users could easily click on delete when what they wanted was to view extra details of it.

However, like any other incumbent organization, the Bank has its own pace when it comes to embracing new ways of thinking. If UX is to be implemented across all the Bank's divisions in the future, its usefulness must first be recognized. It seems very pertinent to start at the Central Bank's OSDD, in charge of in-house operating computer systems development[2]. I have worked for this division for almost ten years, and I hope that the skills I have acquired during that time will allow me to spot UX opportunity areas more easily.

Two questions I aim to answer by conducting the research associated with this thesis are: 1) What is the current UX level of the operating computer systems in the Bank? and 2) Is the UX level indicated by users consistent with the way developers see it?

In chapter 2, I will review different authors that touch upon UX and their different but converging points of view regarding how UX should be defined and modeled. The scope of UX will also be addressed as well as the difficulties that industry practitioners face when trying to account for UX requirements. This chapter is not meant to be an exhaustive literature review on UX but rather a general overview of what UX is according to different researchers in the field.

Chapter 3 presents a UX framework proposal on the basis of which I will measure the UX level of operating computer systems in the Bank. This framework is a result of literature review on UX to categorize its main aspects (e.g., ergonomic quality, hedonic quality) as well as to define the variables that describe it (e.g., security, accessibility, innovativeness, aesthetics). This framework will lay the foundation for all the work that will be done in the following chapters.

Chapter 4 will address the design of two surveys whose aim is to help measure the UX level of the computer systems built in the OSDD. Both surveys' design will be based on the UX framework from Chapter 3. I will outline the main framework characteristics that were used for designing both surveys and will review their structure. Additionally, I will explain the process for refining both surveys in order to make them clearer and more

[2] The operating computer systems are used by the front and back offices for holding operations in both local and international markets.

concise. Finally, I will describe the methods that will be used to do the analysis of surveys' results.

In chapter 5, I will analyze the surveys' results and present statistics that will shed light on the main UX opportunity areas in the Bank.

In Chapter 6, I will discuss the spotted UX opportunity areas and give some recommendations for improvement.

Finally, in chapter 7, I will summarize this thesis and discuss possible next steps, both for the Bank and for future research.

Chapter 2. Literature Review

User Experience (UX) has become a buzzword these days. It is regarded as something valuable but not always easy to understand. One might have heard people say that a particular product delivers a great user experience without consciously understanding what that really meant. In the software industry, UX and usability tend to be regarded as the same thing, implying that a system designed to be usable also offers a satisfactory UX. I have to admit that before I started digging into the UX field, I considered that both terms could be interchangeable, but something kept telling me that there should be a difference between them.

At present, there is no international consensus regarding the reach of the UX definition. Some believe that the UX term should only be used when referring to experiences emerging from the use of products, systems, services, and objects but not when talking about arts, brands, events, etc. [3]. However, this might be confusing since products, services, objects, arts, brands, and events could all be regarded as systems from a purist perspective. For me, any system (whatever its nature) that interacts with a human user delivers a user experience.

Definitions of UX

Let us analyze some definitions of UX that include points of view from some researchers and organizations in the field.

The International Organization for Standardization (ISO) defines UX as: “A person’s perceptions and responses resulting from the use and/or anticipated use of a product, system, or service” [2]. This definition implies that UX can be seen as something temporal that can happen before or during the use of a system, which makes sense, considering that we as humans can experience feelings in stages. For example, before going to see a movie, we start experiencing feelings such as excitement and start raising or lowering our expectations about the film, which could be regarded as the pre-experience. Then, during the film, our expectations are either fulfilled or not and we can experience satisfaction or disappointment with the use of this product or service —

in this case, the film watching. What this definition does not include is a post-experience stage. Some people become huge fans of films to the point that they are looking forward to seeing the next version of it. That is, they keep experiencing emotions even after stopping their direct interaction with the product.

Kashfi, Nilsson, and Feldt state the following: “UX is a user’s holistic perception of functionality and quality characteristics of a piece of software” [4], a definition they based on [5] [6] [7]. The word *holistic* denotes a sense of a whole, that is, all the features of a product that combine together and create an overall perception of it. If we think of the experience a product delivers, it makes sense that all its parts are involved in its overall appraisal: the way the product looks, the functionalities the product has, the way one feels when holding it or looking at it, etc. When thinking of a whole one could argue that functionality and usability (i.e., the do-goals) might not completely define user experience but rather just a part of it, leaving room for a whole range of other aspects such as the hedonic ones (those more emotion-related).

In a video [8] where Don Norman was asked to define user experience, he stated: “It’s everything. The way you experience the world, the way you experience your life — it’s the way you experience a service, or yeah, an app or a computer system, but it’s a system, it’s everything.” This is a more system-thinking approach to a UX definition where a product is seen as a system made of different parts that interact with one another to create a whole that is more than the sum of its parts. The term “holistic” is implicit in Norman’s definition and implies that if one is to care about UX, it becomes crucial to take into account all aspects of the experience.

The Interaction Design Foundation defines UX design as: “the process of creating products that provide meaningful and relevant experiences to users. This involves the design of the entire process of acquiring and integrating the product, including aspects of branding, design, usability, and function” [19]. This definition addresses usability and function as part of UX design but it also emphasizes aspects like branding and findability (acquiring). That is, the way systems are sold to users, how products can be attractive enough to users so that they acquire them, and how easy it is for users to have access to them or become aware of their existence.

When developing computer systems, engineers tend to focus on the technical aspect of them and often forget that they are building products that will make a first lasting impression on users. Yet they tend to see this as superfluous and meaningless. When it comes to in-house systems development, the common thinking is that developed products (e.g., computer systems) are not subject to external competition and this is often regarded as no competition at all. This causes engineers to overlook aspects that might confer a competitive advantage to their products, such as branding. In the computer systems terrain that might include things such as logos, homogeneous Graphical User Interfaces (GUIs), and the access to a standard set of features across all in-house developed software. This might help to establish an internal branding that would end up reflecting a set of quality aspects that in-house products offer regardless of which they use.

Why bother, one might argue, if users that belong in an organization will be forced to use in-house products anyway? It is hard for engineers to see that they are competing against themselves in the first place and that over time in-house developed systems might become external outsourced developed systems under certain circumstances. Furthermore, let us not forget about competition within the organization, where other areas or divisions might take over the design and implementation of products, leaving the original division behind and obsolete.

If a division within an organization or company is to retain an internal captive market, engineers should start paying attention to developing products that account for non-evident aspects. These products should be flexible enough to evolve at the same pace commercial products do if they are to keep users happy and willing to continue using them. No evident competition should not be a reason for no further improvement, and no competition now does not mean no competition in the future.

Marc Hassenzahl [1] divides UX into two fundamental aspects: do-goals and be-goals. While the do-goals focus on the functionality and usability of products, be-goals focus more on the reasons behind the use of these products. Be-goals are more related to why people undertake certain actions or execute certain tasks. For example, Facebook could be thought of as a product that incorporates a wide range of options for doing social networking such as stories and the wall that provides the latest news about your friends. A do-goal on Facebook could be *writing a post* and its corresponding be-goal

could be *stay relevant to friends.* The be-goal, in this case, comes first and triggers the do-goal, but without the *writing a post* capability, the be-goal could never be accomplished. In this case, Facebook as a product manages to provide the do and be capabilities, thus allowing for a more complete user experience. Writing posts on a digital wall is fine but it is better when your friends see them and even better when they "like" them.

UX be-goals show a dimension that goes beyond the fact of being able to do or perform things. In [4] a UX designer that was interviewed mentioned: "*if I start with 'how', I will never get to the 'why'. If I start with 'what', with just making things, I will totally miss every important point there is. For me it tends to be very, very useful to focus on the why. So if I can sort of see this why, even if it's very, very unclear, I can sort of approach this 'how' and 'what' in a much better way*." This quote implies that there is always an underlying reason why users want to have a system built and it is developers' responsibility to figure it out if they are to deliver systems that account for the be part of the whole.

It is important to know what users' real needs and everyday struggles are, but it is also important to understand how they do their work on a daily basis, so programmers can get to the why and deliver systems that better adapt to users' mental models. For example, users in the Central Bank of Mexico might want to automate a process for detecting market makers' bad practices in order to make the process more efficient and effective. In this case, it is important for developers to know what the consequences of violations committed by market makers might lead to and how users do this process right now. Once developers understand the why behind a user's automation request, they will be more willing to incorporate be-goals (i.e., interfaces that account for users' mental models) in their implementation.

Programmers that implement do-goals without accounting for be-goals usually tend to hit a wall when it comes to users' approval of a system. Users might feel their expectations were not met and see the final product as limited, unfamiliar, and ultimately unfriendly. If a system's roll-out depends entirely on users' final approval (i.e., beta testing) engineers that did not account for be-goals could face their system's rejection. This might lead engineers to feel frustrated and wonder why users cannot see how well the system performs and all the effort and creativity they have put into it. They

might even blame users by saying “they do not really know what they want” or “they reject it because they do not understand it”, etc.

Characteristics of UX

Definitions of UX showcase it as something holistic. This might be overwhelming for developers or designers that are trying to incorporate it into their daily work as it is not clear what aspects UX really covers just by looking at its definition. Hence, a more specific approach towards explaining the reach of UX becomes crucial if we are to understand its real implications.

UX expert Peter Morville [9] uses seven factors to describe UX: useful, usable, findable, credible, desirable, accessible, and valuable. Let us see what these factors mean.

Useful. A system must serve a purpose; otherwise, it will not be useful. Before engaging in new system design and implementation, one should ask oneself if what is going to be built will help the intended users accomplish something that is relevant for them. A system that serves no purpose is a system that will not be used at all.

Usable. Once a user need is spotted and a computer system seems to be the right way of fulfilling it, programmers should be concerned about bringing a usable interface to users. Even if a system is relevant to users, they will not use it or will end up making less use of it if its interface is unfriendly and does not fit their mental models. It is crucial then to build interfaces that are intuitive and communicate the right affordances[3] to users if they are to embrace them.

Findable. Users will not buy or use a product or system they cannot find and even if they discover how to get access to it, they still have to be able to easily spot the features offered by it. From my own experience, developers do not want to work on features users will not use, yet some get blinded by their own initiative and add functionalities without letting users know in advance in hope that these will somehow be discovered and used accordingly. One can imagine that this unlikely to happen and, in

[3] The possible actions users can take within a system [10].

some cases, users can become frustrated when realizing they cannot do their work due to the lack of a specific feature. When asked why a system does not have specific functionality that actually it does provide, programmers might be tempted to think users just do not understand the interface. However, developers should ask themselves if such features are discoverable in the first place.

Credible. Before they start interacting with a system (e.g., application, website, car, plane, etc.) users have to build a certain level of trust in it. This trust encompasses many aspects such as robustness, information integrity, availability, security[4], scalability, etc. Some aspects will be more important than others, depending on the system's purpose, and on the system's level of credibility from the user's perspective.

Desirable. Desirability can be interpreted in many ways. One can desire something because it is trendy, aesthetic, original, alleviating, exciting, etc. Products that are desirable are more likely to be acquired or adopted by users. When building software, programmers should account for desirability by highlighting those aspects that would make their products stand out from the rest. When talking about software desirability in the Bank, programmers might particularly account for aesthetics, innovativeness, and originality. Excitement might be a desirable characteristic too, but harder to achieve due to people's perspectives about banking software, not always considered as something that has to be exciting, compared to the software made in the gaming industry.

I have a vivid memory from years ago when the OSDD rolled out a new version of a system designed for conducting governmental securities auctions. After the auction ended and the analyst in charge released results privately (i.e., to institutions that were assigned securities) and publicly (i.e., to the Internet) the manager said out loud in a very happy and pensive way, "Can you think of all the work that is behind this system in order for us to be capable of pushing a single button and have everything processed and sent to institutions in a blink of an eye?" I felt very proud of having been involved in

[4] Security covers many aspects such as secure interfaces (e.g., wrong buttons or options in the interface are easily selectable) and information security (e.g., unauthorized access to privileged information). When talking about UX, both are important. However, information security is not something users can easily perceive and when they do it is usually because a security breach has occurred thus making evident a system's security hole. Normally in UX, interface security will be the aspect to evaluate.

that system and, of course, I saw nothing more than excitement for how the system facilitated my colleagues' work.

I firmly believe that excitement can emerge from any computer system, whatever its nature is. Of course, that excitement it will show in a subtler way in systems like those used in central banking or scientific research (e.g., physics, math, computer science, mechanical engineering) than in those used in the gaming, design, or music industries.

Accessible. Accessibility is something very few people think about unless they belong or have a relative that belongs to the 12.8% of the population with some kind of disability in the US [11] or to the 5.1% in Mexico[5] [12]. People with disabilities have a hard time every day accessing different kinds of places such as websites and buildings or services such as public transportation and education. When it comes to software, people with disabilities can face challenges such as problems differentiating red, green, blue, and yellow colors (i.e. color-blindness), or hearing difficulties.

Imagine an analyst who has color blindness trying to interpret a dashboard packed with red and green colors (e.g., economic indicators that show negative numbers in red and positive ones in green for calling attention more easily) or a person with hearing disabilities operating a system that sends auditory alarms when a critical process has finished its execution or when something has gone wrong. These users will get frustrated very easily and, in some cases, they might be blamed for something out of their reach. If we are to be inclusive, we have to know our users and build systems that consider their special needs.

In the Bank, more accessible software might incorporate tools that allow people to have a customized interface according to their needs. For example, the first time a certain software is opened, it might run a simple test to determine if a person has color blindness [13]. If the test runs positive, the interface might adjust automatically in order to allow this person to work normally. The same could apply for people with hearing problems: the system could automatically switch to visual aids instead of auditory alarms.

[5] This is the official number. Perhaps the real one is higher.

Valuable. A system that does not deliver value to a company or organization does not have a reason to be in the first place. Software must deliver value, or potential users will not even look at it or advocate for its development. In the Bank, the software helps to conduct sound monetary and exchange policies for the country, thus delivering a more stable economy where financial markets can thrive. Before undertaking a software development project, one must know what value will emerge from it. If the value emerging is not clear, then we should think again whether it is worth pursuing.

Morville's factors are a good starting point towards understanding the characteristics a product should have in order to account for UX as a whole. An example of a product designed taking into account these aspects is the iPhone, which is easy to use (it is usable), desirable (thanks to its aesthetics, good reputation, and trendiness), accessible (it is designed for every person, including people with visual, hearing, physical, and motor disabilities), useful (it helps to accomplish many daily life tasks regardless of their nature: leisure, professional, personal finances, news, health, weather, etc.), findable (information about it is easy to retrieve), valuable (it generates value for people who use it for either personal or professional reasons), and credible (most of its users trust and rely on it).

Another perspective that contributes towards defining UX is the one from Hassenzahl [14] who considers that UX has two traits: ergonomic quality (e.g., efficiency, accessibility, learnability) and hedonic quality (e.g., originality, innovativeness). These qualities are perceived independently by users [15] and combine in their minds to form an appeal of a system, product, or service. The possibility of an emergent appeal formed by a mix of both qualities raises the concept of a trade-off between ergonomic and hedonic elements that results in a final user's appraisal.

In [5], Hassenzahl proposes a UX model that accounts for both designer and user perspectives. From the designer's angle, UX is made out of product features (e.g., content, presentation, functionality, and interaction), which are supposed to lead to an intended product character that considers pragmatic (i.e., ergonomic) and hedonic attributes.

This intended character will project a certain product's appeal and elicit certain emotions from users such as excitement or happiness. When users get in touch with the product, they construct an apparent product character with their own appreciation of pragmatic and hedonic attributes that might differ from the designer's original intention. This particular appreciation leads in turn to consequences such as appeal and emotions that again might be different than the ones intended by the designer. An important takeaway of these models in terms of software construction is that one must differentiate between what developers think UX will be and what users actually perceive. Programmers must do their best to ensure both perspectives are in harmony as much as possible.

Mahlke, Sascha, and Manfred [16] propose the Components of User Experience (CUE) model that encompasses three components: instrumental qualities, non-instrumental qualities, and emotional reactions. Instrumental quality is related to features offered by a system and its usability whereas non-instrumental qualities touch upon the look and feel of the system. Both qualities combine and generate emotions. Ultimately, the combination of instrumental qualities, non-instrumental qualities, and emotions results in a user's appraisal of a system. This way of explaining UX is very similar to the one used by Hassenzahl in that pragmatic attributes map to CUE's instrumental qualities, and hedonic attributes to CUE's non-instrumental qualities.

The main difference between Hassenzahl's model [5] and the CUE-model is that the latter considers that emotions emerge first as a result of a user's perception of both qualities (i.e., instrumental and non-instrumental) and then recombine again with the qualities to generate a user's appraisal. In contrast, Hassenzahl's model states that both qualities (i.e., ergonomic and hedonic) combine and then result in a product's appeal and the user's emotions. Figures 1 and 2 illustrate this.

All proposed UX models have elements that intersect and support a common assertion: UX is more than just usability or instrumental quality and its reach includes hedonic or non-instrumental qualities that relate to non-functional features of systems (products, services, and objects can all be treated as systems). Both instrumental and non-instrumental qualities combine and result in user emotions and ultimately in an overall user's appraisal of a system.

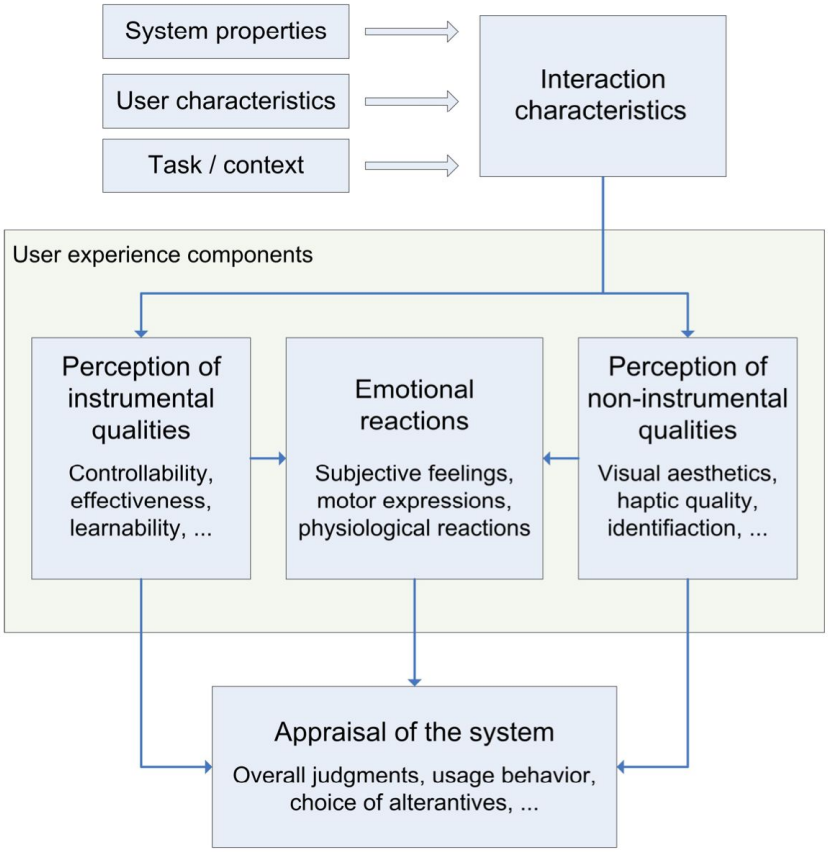


Figure 1. Mahlke, Sascha, and Manfred CUE-model [16].

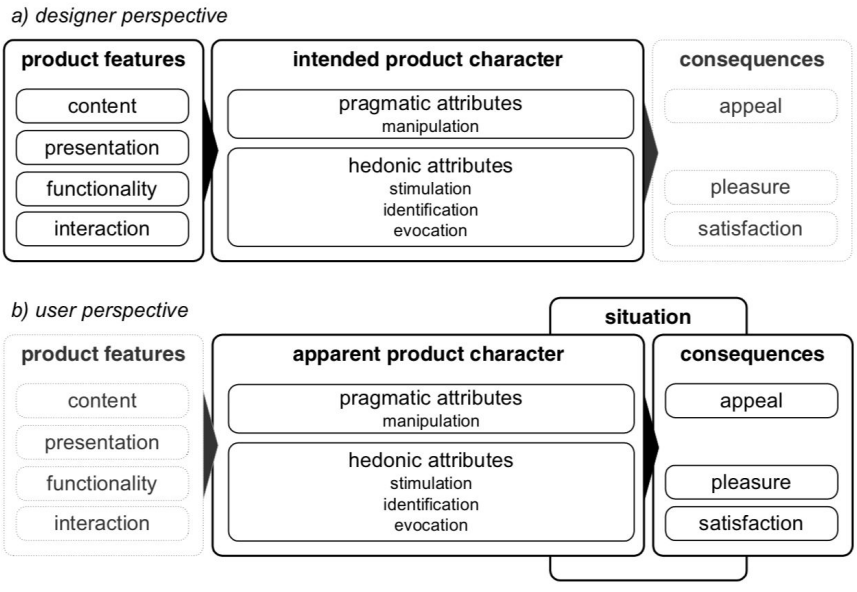


Figure 2. Hassenzahl UX model with a) designer and b) user perspectives [5].

Table 1 summarizes all approaches from models that were analyzed and their intersections. All models seem to have three main parts: 1) Objective (e.g., functionality and usability), 2) Subjective (e.g., innovativeness, aesthetics), and 3) Emergence (what results when objective and subjective parts interact with one another).

Table 1. Studied UX models and their intersections.

Author	Objective	Subjective	Emergence
Hassenzahl (UX Qualities) [14]	**Ergonomic Quality** (EQ) (Functionality and Usability)	**Hedonic Quality** (HQ) (Refers to the quality dimensions with no obvious or at least a second order relation to task-related goals such as originality, innovativeness, and so forth)	Appeal/ Attractiveness
Hassenzahl (UX Attributes) [5]	**Designer perspective** (intended product character) and **User perspective** (apparent product character)		Consequences
	Pragmatic (usability and functionality of software) Manipulation	**Hedonic** (Communicating identity, provoking memories, and providing stimulation) Stimulation, Identification, and Evocation	Appeal, Pleasure, Satisfaction
Hassenzahl (Be & Do goals) [1]	Do-goals	Be-goals	-
Mahlke, Sascha, and Manfred (CUE-Model) [16]	**Instrumental Qualities** (Usability and Usefulness) Controllability, Effectiveness, Helpfulness, & Learnability	**Non-Instrumental Qualities** (Appeal and attractiveness) Look & Feel: Visual aesthetics or haptic quality	Emotional reactions
Peter Morville (UX factors) [9]	Usable, Findable, Accessible, Credible	Desirable, Valuable, Useful	-

The Scope of UX

In 2009, Hassenzahl et al. [17] did research whose aim was to define UX and its scope. They conducted an online survey of a number of UX researchers and UX practitioners from different industries in order to get a sense of their level of agreement with a set of UX statements and collect their opinions on one or more previously chosen UX definitions. Regarding the scope of UX, the authors emphasized the difference between brand experience, product experience, and service experience.

Brand experience is seen as something broader than UX since it includes every single interaction users might have with a brand which might, in turn, affect the user experience of people who interact with the product [17]. Every TV advertisement we have seen about a brand, what people have told us about their own experiences with it, etc., affect the brand experience from the users' perspective. A good brand experience can manage to keep us captive even if sometimes brands do things we dislike (e.g., child exploitation, environmental pollution).

If the subjectivity of UX is to be emphasized, then *product experience* should be regarded as something narrower than UX. That way, we will not be tempted to say "a product delivers a certain UX" instead of "a person has a certain UX when using a product." The authors of the study stated that individuals can share with others the emotions that result from their experiences using a certain product, but the experiencing process itself is restricted to the personal level [17]. This statement implies that UX can be seen as *subjective*, as only individuals can have experiences.

Finally, Hassenzahl et al. [17] state that since services might be of different nature (e.g., face-to-face services, digital services, public services) every type should be evaluated on a case basis to determine whether or not it is within the focus of UX. Regarding *service experience*, the authors argue that face-to-face services are out of UX scope since humans do not have a user interface, that is, we cannot "use" humans. I slightly disagree with this argument since humans do have an interface through which they communicate with other humans. Spoken or sign languages are some examples of such interfaces and humans can use them to manipulate or use other humans (e.g., blackmail, communication of feelings in order to move someone to do something for us, spoken instructions to somebody in order to accomplish a task).

Human interfaces might not be tangible, but their mobilizing power cannot be denied. However, if we are to restrict the UX field's scope we had better stick to those interfaces that were specifically and artificially designed to be utilized by users. In that case, face-to-face interactions would actually be left out of UX scope.

Since we can have a wide variety of experiences, it becomes crucial to define the boundaries of UX. Hassenzahl et al. [17] proposed that we think of UX as the experience we have via user interfaces, thus leaving out all experiences that emanate from interfaces other than those explicitly created to be manipulated by users. Figure 3 illustrates this idea.

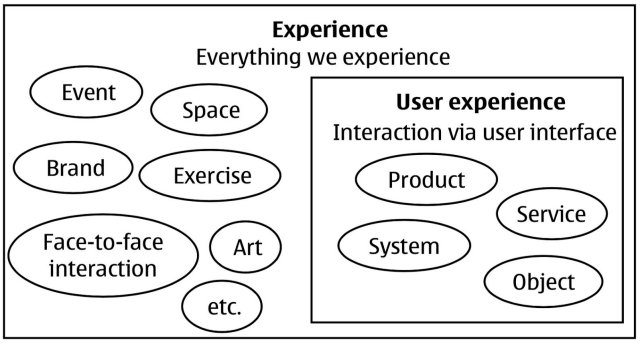


Figure 3. UX in relation to other experiences that can be studied (Hassenzahl et al. [17]).

We can experience art. However, light which acts as a natural interface between our eyes and paintings or sculptures was not artificially created to be manipulated by humans. An example of art one might think could actually be considered within the focus of UX is music deliberately composed to make people do certain things such as stay awake, concentrate better, or enter a deep phase of sleep more quickly. In this case, there is an interface that was artificially created and that can be manipulated by humans to make themselves or other humans perform certain things. The only reason why manipulative music would be out of UX scope as previously defined is that it is not tangible[6]. I hope this explanation can help the reader understand the scope of UX and the subtleties (e.g., artificially created, tangible) that lie behind it in order for him/her to be able to separate UX from other experiences.

[6] By tangible I mean, perceptible by touch.

Chapter 3. A User Experience Framework Proposal

The overarching questions that motivated this thesis work are: 1) What is the current UX level of the operating computer systems in the Bank? 2) Is the UX level indicated by users consistent with the way developers see it?

In order to answer those questions, I see the need to come up with a comprehensive framework that can include all aspects that will be considered when measuring UX. I will make use of some elements from the literature review as well as other bibliographic materials from authors that have already tried to measure UX to suggest a framework that can incorporate all the UX approaches here discussed. I will then apply this framework to measure the operating computer systems current UX level in the Bank (i.e., systems developed by the OSDD).

Preamble to Framework Construction

From all models discussed in chapter 3, I believe the one from Hassenzahl [14] has the clearest terms for defining the parts of UX: ergonomic and hedonic qualities. Both words can be easily found in other contexts (e.g., ergonomic furniture, hedonic life) and one can easily infer their meaning within the UX context. At the same time, this model keeps things simple by stating that the user appeal of a system is the result of his/her interaction with both qualities that represent such a system. Basically, if we measure these qualities to determine an overall UX level and then measure the overall user appeal independently, both should be positively correlated.

But first, a method for qualitatively measuring UX level has to be settled on. One can measure usability through a wide array of metrics such as the number of tasks successfully completed (i.e., effectiveness), the keystroke-level model[7] (i.e., efficiency), etc. In contrast, measuring hedonic aspects of the UX is not so evident. Laugwitz et al. [18] suggest the creation and evaluation of a UX questionnaire for quickly and

[7] In human-computer interaction, the **keystroke-level model** (**KLM**) predicts how long it will take an expert user to accomplish a routine task without errors using an interactive computer system ("Keystroke-Level Model." In *Wikipedia*, February 20, 2019. https://en.wikipedia.org/w/index.php?title=Keystroke-level_model&oldid=884216822.)

comprehensively evaluating a product's UX (i.e., ergonomic and hedonic qualities as well as appeal).

Laugwitz et al. [18] built their questionnaire based on a theoretical framework of UX [15]. They then used an empirical approach for selecting the items that most represented UX. This consisted of selecting a group of fifteen SAP[8] usability experts that came up, after some brainstorming, with the most important terms of UX in their own experience. Then a down-selection process was made over these terms to remove duplicates and keep only the most relevant ones. The final list of terms resulted in the construction of a 26-item questionnaire. Additionally, the items were classified into six different categories: Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, and Novelty.

Attractiveness terms would be directly related to *Appeal,* whereas Perspicuity, Efficiency, and Dependability terms would be related to *Ergonomic Quality* (EQ) and Stimulation and Novelty ones to *Hedonic Quality* (HQ). All these terms represented adjectives related to the way UX was being evaluated within SAP. The questionnaire for measuring UX was based on the semantic differential method[9]. Hence, antonyms for each adjective were identified in order to be able to use a polar scale to collect qualitative data from users. Authors of the questionnaire decided to use a seven-point scale (shown in Figure 4) to reduce central tendency bias.

Figure 4. Seven-point scale used in Laugwitz et al. [18] questionnaire.

Table 2 summarizes the list of UX terms or adjectives under their specific category.

[8] **SAP** ("**S**ystems, **A**pplications & **P**roducts in Data Processing") is a German-based European multinational software corporation that makes enterprise software to manage business operations and customer relations ("SAP SE." In *Wikipedia*, March 26, 2019. https://en.wikipedia.org/w/index.php?title=SAP_SE&oldid=889523956.)

[9] **Semantic Differential** (**SD**) is a type of rating scale designed to measure the connotative meaning of objects, events, and concepts. The connotations are used to derive the attitude towards the given object, event or concept. ("Semantic Differential." In *Wikipedia*, February 21, 2019. https://en.wikipedia.org/w/index.php?title=Semantic_differential&oldid=884382807.)

Table 2. Extract of terms used for constructing Laugwitz et al. User Experience Questionnaire (UEQ) [18].

Category	Polar adjectives	
Attractiveness	annoying	enjoyable
	attractive	unattractive
	friendly	unfriendly
	good	bad
	unlikable	pleasing
	unpleasant	pleasant
Dependability	obstructive	supportive
	secure	not secure
	unpredictable	predictable
	meets expectations	does not meet expectations
Efficiency	fast	slow
	impractical	practical
	inefficient	efficient
	organized	cluttered
Novelty	conservative	innovative
	creative	dull
	inventive	conventional
	usual	leading edge
Perspicuity	clear	confusing
	complicated	easy
	easy to learn	difficult to learn
	not understandable	understandable
Stimulation	boring	exciting
	motivating	demotivating
	not interesting	interesting
	valuable	inferior

As I mentioned previously, Laugwitz et al.'s [18] questionnaire construction was based on Marc Hassenzahl's framework [15] which also proposes the use of a semantic differential questionnaire for measuring EQ, HQ, and Appeal. Hassenzahl came up with the list of polar adjectives shown in Table 3 and also made use of a seven-point scale for collecting users' perceptions.

Table 3. Bipolar adjectives or Anchors used for constructing Hassenzahl [15] UX questionnaire.

Aspect to measure	**Anchors**	
Ergonomic Quality (EQ)	Comprehensible	Incomprehensible
	Supporting	Obstructing
	Simple	Complex
	Predictable	Unpredictable
	Clear	Confusing
	Trustworthy	Shady
	Controllable	Uncontrollable
	Familiar	Strange
Hedonic Quality (HQ)	Interesting	Boring
	Costly	Cheap
	Exciting	Dull
	Exclusive	Standard
	Impressive	Nondescript
	Original	Ordinary
	Innovative	Conservative
Appeal	Pleasant	Unpleasant
	Good	Bad
	Aesthetic	Unaesthetic
	Inviting	Rejecting
	Attractive	Unattractive
	Sympathetic	Unsympathetic
	Motivating	Discouraging
	Desirable	Undesirable

I decided to use the polar adjectives lists of Laugwitz et al. [18] and Hassenzahl [15] in order to come up with a new list that would incorporate the best of both approaches.

I used a DSM[10] approach to easily compare all adjectives from both lists in order to spot duplicates (i.e., synonyms) and terms whose meaning might overlap or be slightly related between both lists. Figures 5, 6, and 7 show this process for adjectives within EQ, HQ, and Appeal, respectively. The orange region corresponds to Laugwitz et al. and the blue one to Hassenzahl. The white region corresponds to two adjectives I took from Morville's characterization of UX [9] that had not been considered by [18] and [15] but which I see as crucial for a complete UX evaluation. Hassenzahl does not have specific categories for his adjectives originally and that is why I reclassified them using Laugwitz et al. categories for keeping consistency. I did the same for Morville's adjectives.

Category	Adjective	Meets expectations/Doesn't meet expectations	Secure/Not secure	Obstructive/Supportive	Unpredictable/Predictable	Not Understandable/Understandable	Accessible/Inaccessible	Findable/Not findable	Easy to learn/Difficult to learn	Complicated/Easy	Clear/Confusing	Fast/Slow	Inefficient/Efficient	Impractical/Practical	Organized/Cluttered	Comprehensible/Incomprehensible	Supporting/Obstructing	Simple/Complex	Predictable/Unpredictable	Clear/Confusing	Trustworthy/Shady	Controllable/Uncontrollable	Familiar/Strange
Dependability	Meets expectations/Doesn't meet expectations	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Secure/Not secure	0	X	0	1	0	2	0	0	0	1	0	0	0	0	0	0	0	1	1	2	2	0
	Obstructive/Supportive	0	0	X	1	1	2	0	1	1	1	0	1	2	0	0	3	0	1	1	0	1	0
	Unpredictable/Predictable	0	1	1	X	0	0	0	0	0	0	0	0	0	1	0	1	1	3	0	1	0	0
	Not Understandable/Understandable	0	0	1	0	X	0	0	1	3	3	0	0	0	0	3	0	0	0	3	0	0	2
Dependability	Accessible/Inaccessible	0	2	2	0	0	X	0	0	1	0	0	0	0	0	0	2	0	0	0	0	0	0
	Findable/Not findable	0	0	0	0	0	0	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Perspicuity	Easy to learn/Difficult to learn	0	0	1	0	1	0	0	X	2	2	0	0	0	0	2	0	2	0	2	0	0	2
	Complicated/Easy	0	0	1	0	3	1	0	2	X	2	0	0	0	0	2	1	2	0	2	0	0	2
	Clear/Confusing	0	1	1	0	3	0	0	2	2	X	0	0	0	0	2	1	1	0	3	0	0	1
Efficiency	Fast/Slow	0	0	0	0	0	0	0	0	0	0	X	3	0	0	0	0	0	0	0	0	0	0
	Inefficient/Efficient	0	0	1	0	0	0	0	0	0	0	3	X	0	0	0	0	0	0	0	0	0	0
	Impractical/Practical	0	0	2	0	0	0	0	0	0	0	0	0	X	1	1	1	1	0	1	0	0	0
	Organized/Cluttered	0	0	0	1	0	0	0	0	0	0	0	0	1	X	1	1	1	0	2	0	1	0
Dependability	Comprehensible/Incomprehensible	0	0	0	0	3	0	0	2	2	2	0	0	1	1	X	1	1	0	3	0	0	1
	Supporting/Obstructing	0	0	3	1	0	2	0	0	1	1	0	0	1	1	1	X	1	1	1	0	1	0
Perspicuity	Simple/Complex	0	0	0	1	0	0	0	2	2	1	0	0	1	1	1	1	X	1	2	0	0	1
Dependability	Predictable/Unpredictable	0	1	1	3	0	0	0	0	0	0	0	0	0	0	0	1	1	X	0	0	1	0
Perspicuity	Clear/Confusing	0	1	1	0	3	0	0	2	2	3	0	0	1	2	3	1	2	0	X	0	0	1
Dependability	Trustworthy/Shady	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X	0	0
	Controllable/Uncontrollable	0	2	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	X	0
Perspicuity	Familiar/Strange	0	0	0	0	2	0	0	2	2	1	0	0	0	0	1	0	1	0	1	0	0	X

Figure 5. EQ polar adjectives resemblance comparison.
Keys: 0=Totally differentiable, 1=Slightly related, 2=Overlapping, 3=Synonyms, X=Same.

[10] The Design Structure Matrix (DSM) is a compact and visual representation of a system or project in the form of a square matrix. ("Design Structure Matrix." In *Wikipedia*, January 6, 2019. https://en.wikipedia.org/w/index.php?title=Design_structure_matrix&oldid=877072448.)

Category	Adjective	Conservative/Innovative	Usual/Leading edge	Inventive/Conventional	Creative/Dull	Valuable/Inferior	Boring/Exciting	Not interesting/Interesting	Motivating/Demotivating	Interesting/Boring	Costly/Cheap	Exciting/Dull	Exclusive/Standard	Impressive/Nondescript	Original/Ordinary	Innovative/Conservative
Novelty	Conservative/Innovative	X	2	3	2	0	0	0	0	0	0	0	0	0	3	3
	Usual/Leading edge	2	X	3	2	0	1	0	0	0	0	2	0	1	2	1
	Inventive/Conventional	3	3	X	1	1	3	2	0	2	0	1	0	0	3	3
	Modernity - Creative/Dull	2	2	1	X	0	1	1	0	1	0	1	0	0	1	1
Stimulation	Valuable/Inferior	0	0	1	0	X	0	0	0	0	0	0	0	0	0	0
	Boring/Exciting	0	1	3	1	0	X	3	1	3	0	1	0	1	1	1
	Not interesting/Interesting	0	0	2	1	0	3	X	0	3	0	1	0	1	1	1
	Motivating/Demotivating	0	0	0	0	0	1	0	X	0	0	1	0	1	1	1
Stimulation	Interesting/Boring	0	0	2	1	0	3	3	0	X	0	2	1	2	1	1
	Costly/Cheap	0	0	0	0	0	0	0	0	0	X	0	0	0	0	0
	Exciting/Dull	0	2	1	1	0	1	1	1	2	0	X	1	2	1	1
	Exclusive/Standard	0	0	0	0	0	0	0	0	1	0	1	X	0	1	0
	Impressive/Nondescript	0	1	0	0	0	1	1	1	2	0	2	0	X	0	0
Novelty	Original/Ordinary	3	2	3	1	0	1	1	1	1	0	1	1	0	X	2
	Innovative/Conservative	3	1	1	1	0	1	1	1	1	0	1	0	0	2	X

Figure 6. HQ polar adjectives resemblance comparison.
Keys: 0=Totally differentiable, 1=Slightly related, 2=Overlapping, 3=Synonyms, X=Same.

Category	Adjective	Enjoyable/Annoying	Good/Bad	Unlikable/Pleasing	Unpleasant/Pleasant	Attractive/Unattractive	Friendly/Unfriendly	Pleasant/Unpleasant	Good/Bad	Aesthetic/Unaesthetic	Inviting/Rejecting	Attractive/Unattractive	Sympathetic/Unsympathetic	Motivating/Discouraging	Desirable/Undesirable
Appeal	Enjoyable/Annoying	X	1	3	2	0	0	2	0	0	0	0	0	0	0
	Good/Bad	1	X	0	0	0	0	0	3	0	1	0	0	0	0
	Unlikable/Pleasing	2	0	X	3	1	0	3	0	0	0	0	0	0	0
	Unpleasant/Pleasant	2	0	3	X	1	0	3	0	0	0	0	0	0	0
	Attractive/Unattractive	0	0	1	1	X	0	0	1	0	2	3	1	0	1
	Friendly/Unfriendly	0	0	0	0	0	X	1	0	0	2	0	3	0	0
Appeal	Pleasant/Unpleasant	2	0	3	3	0	1	X	0	0	0	0	1	0	0
	Good/Bad	0	3	0	0	1	0	0	X	0	0	0	0	0	0
	Aesthetic/Unaesthetic	0	0	0	0	0	0	0	0	X	2	3	0	0	0
	Inviting/Rejecting	0	1	0	0	2	2	0	0	2	X	3	0	0	0
	Attractive/Unattractive	0	0	0	0	3	0	0	0	3	3	X	1	0	1
	Sympathetic/Unsympathetic	0	0	0	0	1	3	1	0	0	0	1	X	0	0
	Motivating/Discouraging	0	0	0	0	0	0	0	0	0	0	0	0	X	0
	Desirable/Undesirable	0	0	0	0	1	0	0	0	0	0	1	0	0	X

Figure 7. Appeal polar adjectives resemblance comparison.
Keys: 0=Totally differentiable, 1=Slightly related, 2=Overlapping, 3=Synonyms, X=Same.

After doing the resemblance comparison, I applied the following criteria for shortening the list of adjectives:

1. From all synonyms (i.e., number 3), I removed duplicates when necessary. From those that remained I kept the ones easier to translate to and understand in Spanish according to my own experience as a native speaker;
2. I then tried to think of whether those in the remaining set really differentiated from each other or not. It turned out that some did not (i.e., Trustworthy/Shady whose meaning could be confused with Secure/Not secure), hence I chose the most appealing ones and left out the rest;
3. I removed those adjectives not relevant for the Bank's in-house computer systems (i.e., Costly/Cheap[11], Sympathetic/Unsympathetic);
4. From those that duplicated or overlapped across UX dimensions (i.e., EQ, HQ, and Appeal) I decided to keep the ones that were more likely to pertain to a certain dimension (i.e., Practical/Impractical from EQ might have overlapped with Friendly/Unfriendly from Appeal; Motivating/Discouraging seemed to be more related to HQ than Appeal); and finally
5. I decided to change some adjectives for others with a similar interpretation in the field of usability (i.e., Familiar/Strange for Consistent/Inconsistent).

In the case of Consistent/Inconsistent polar adjectives, I decided to measure both internal and external consistency. Since I used a heuristic approach for coming up with a more compact list, I do not expect it to be perfect, but rather useful for the purpose of defining a new framework that will try to employ the best of the approaches here discussed. The final list containing 31 pairs of polar adjectives along with their UX dimension, category, and representative noun is shown in Table 4.

[11] Our users do not have to pay to get the computer systems they need built. The Bank has decided to keep an in-house team for doing that job due to the high sensitivity of information.

Table 4. The final list of polar adjectives classified by UX Dimension and Category.

UX Dimension	Category	Noun	Polar adjectives	
Ergonomic Quality (EQ)	Dependability	Expectation	Doesn't meet expectations	Meets expectations
		Security	Not Secure	Secure
		Supportiveness	Obstructive	Supportive
		Predictability	Unpredictable	Predictable
		Accessibility	Not Accessible	Accessible
		Findability	Not Findable	Findable
		Controllability	Uncontrollable	Controllable
	Perspicuity	Learnability	Difficult to Learn	Easy to Learn
		Clarity	Confusing	Clear
		Simplicity	Complex	Simple
		Internal Consistency	Inconsistent	Consistent
		External Consistency	Inconsistent	Consistent
	Efficiency	Efficiency	Inefficient	Efficient
		Grouping	Cluttered	Organized
Hedonic Quality (HQ)	Novelty	Innovation	Conservative	Innovative
		Modernity	Usual	Leading-edge
		Creativeness	Dull	Creative
		Originality	Ordinary	Original
	Stimulation	Value	Inferior	Valuable
		Excitement	Boring	Exciting
		Interest	Not interesting	Interesting
		Motivation	Demotivating	Motivating
		Exclusivity	Standard	Exclusive
		Impressiveness	Nondescript	Impressive
Appeal	-	Goodness	Bad	Good
		Likeability	Unlikable	Pleasing
		Attractiveness	Unattractive	Attractive
		Friendliness	Unfriendly	Friendly
		Enjoyment	Annoying	Enjoyable
		Aesthetics	Unaesthetic	Aesthetic
		Desirability	Undesirable	Desirable

UX Framework Proposal

I will base my framework proposal on the work of Laugwitz et al. [18] and Hassenzahl [15], with the latter being the one that I will use as a primary foundation. User Experience then will have three parts or dimensions: 1. Ergonomic Quality (EQ); 2. Hedonic Quality (HQ); and 3. Appeal. This framework assumes that if the UX level of system was measured through its ergonomic and hedonic qualities, this would be positively correlated to its Appeal. In other words, the overall Appeal of a system would be the result of a trade-off between its EQ and HQ measures, which will be determined by the user's perceptions of all pairs of polar adjectives in Table 4. A graphical representation of the proposed UX framework can be observed in Figure 8.

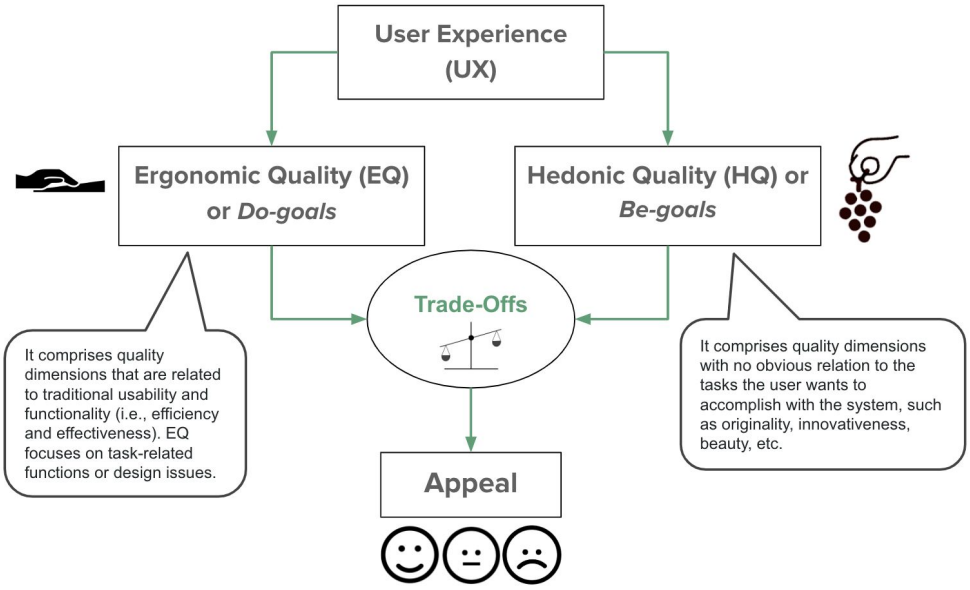


Figure 8. UX Framework proposal.

Polar adjectives could be interpreted in various ways depending on the context in which the reader places them when thinking about them. That is why I will give a definition of what I mean by the noun that represents each pair of polar adjectives in this framework in order to reduce ambiguity when interpreting it.

Definition of Ergonomic Quality (EQ) Polar Adjectives

Dependability

These are features that allow users to depend (rely) on computer systems.

- **Expectation**: This refers to all user expectations a system is intended to fulfill (i.e., functionality). All user requirements are addressed during the system's development process. These are validated (i.e., through integration and beta tests, testing engineers and users verify requirements were met and they accept or reject the system) and verified (i.e., the system complies with regulations, specifications, and special conditions imposed by users)[12].
- **Security**: In this framework, it refers to secure Graphical User Interfaces (GUIs). An example of insecure GUI would be one where it is easy to click on the wrong button thus causing the deletion of important information or the execution of an operation that should not have been processed in the first place. It does not have to do with security breaches that might be caused by hackers exploiting security holes in the computer system. However, this other aspect of security is also important and relevant to the UX.
- **Supportiveness**: This refers to whether the system supports or hinders the user's everyday activities at work. Some systems are built because of internal or external regulations that are usually triggered by users' needs. However, that is not an indication that the way such systems are constructed will help users achieve their work in a more straightforward way.
- **Predictability**: It is desirable that systems are predictable to users. That is, that users know exactly what the consequences will be of executing an action on the GUI (e.g., pressing a button, dragging a component, scrolling down, pressing a sequence of keys).
- **Accessibility**: All kinds of users, regardless of whether they have disabilities (e.g., color blindness, hearing, motor), can interact with the GUI. Interfaces should be prepared to provide an acceptable set of tools depending on the type of disability users might have. It is a good practice to know the audience of a system and plan the accessibility accordingly.

[12] "Verification and Validation." In *Wikipedia*, April 18, 2019. https://en.wikipedia.org/w/index.php?title=Verification_and_validation&oldid=893086834.

- **Findability**: That a system exists does not mean potential users know about it. IT divisions must generate awareness of a system's availability as well as its main functionalities. Additionally, access to a system from its users should be frictionless by providing them with links (e.g., web, icons, menus) to clearly and quickly access all its functionalities.
- **Controllability**: graphic components on the interface can be manipulated easily in order to edit the system's parameters (e.g., dates, figures, ranges) and perform operations (e.g., searching for information, executing processes).

Perspicuity

These are features that facilitate the interaction between users and graphical interfaces.

- **Learnability**: This concerns to how easy a system is to learn, whether users have access to a user manual or not. It is desirable that systems consider users' mental models (i.e., the way users solve problems without a system) in order to lower the interfaces' learning curves.
- **Clarity**: The interface makes use of the appropriate graphical components to present information to users (e.g., lists, tables, dialogues). The representation of information might be clear or confusing depending on the elements used to present it to users. For example, tabular information will look confusing when presented in the form of a list and vice versa.
- **Simplicity**: This refers to how complicated an interface makes a process look. Some interfaces are more transparent than others when it comes to indicating the execution of processes by stages. Some just execute the process without being in touch with users in between, giving the impression that one is working with a black box. It is not easy to understand how something works without having the proper details. This might cause frustration for users that are new to the system and do not yet have the business know-how.
- **Internal Consistency**: a system that has internal consistency uses the same components or graphical representations for conveying functionalities that do the same across all its interfaces or windows (e.g., same icons for print and export buttons wherever they appear on the interface, same way of entering dates, figures, or ranges across all dialogue windows).

- **External Consistency**: Users are in touch with many kinds of interfaces and they get used to spotting certain features by their aspect. For example, it has become a standard to use a printer, a floppy disk, and scissors to represent printing, saving, and cutting actions. A system that is externally consistent will account for known standards for GUI developing. Being externally consistent helps lower the user's learning curve.

Efficiency

These characteristics pertain to the speed that can be attained when using graphical interfaces.

- **Efficiency**: This refers to how quickly a user can execute tasks or processes given a certain interface layout and behavior. Some interfaces allow users to achieve their goals but in a rather slow way (e.g., dates have to be entered manually, figures do not self-format) while others not only are fully functional but also very efficient (e.g., date pickers, drop-down menus for selecting pre-existing options, dynamic option selection that presents only the possible options, thus reducing users' mental effort when selecting as well as potential errors when users have to enter data manually).
- **Grouping**: Things of the same nature that are grouped together, making it easier for users when it comes to interpreting them. In contrast, when same-nature data is scattered through the interface users are forced to increase their level of attention when executing tasks such as looking for the right settings options, interpreting some query results, etc. At the same time, when information is not properly grouped users tend to be more error-prone. The right grouping of information makes interfaces more agile.

Definition of Hedonic Quality (HQ) Polar Adjectives

Novelty

These features refer how users perceive systems' novelty based on their Graphical User Interfaces (GUIs).

- **Innovation**: This refers to the use of new approaches for solving users' everyday

problems at work (e.g., new algorithms that have proven to be faster and as effective as the ones normally used).

- **Modernity**: Systems that keep up with technological trends are usually regarded as modern. Offering features like sharing data with others, preventing users from executing repetitive tasks (e.g., entering their names, addresses, credit card information more than once), allowing group collaboration such as Google docs real-time document edition, and pre-visualizing results before committing an operation, are some examples of what systems regarded as modern offer these days.
- **Creativeness**: Systems that solve problems using ingenious or clever solutions, which in turn require fewer resources and deliver the same results, are regarded as creative.
- **Originality**: Systems that are original make use of tools that were specifically created from scratch (e.g., a graphical component made to fit a specific task such as WhatsApp's following up feature, which allows people to associate their responses to a specific text message previously received, thus making the conversation clearer) to meet users' needs, instead of using standard pre-existing ones that only partially meet users' expectations.

Stimulation

These are factors that can aid users' work or hinder it.

- **Value**: When systems are valuable to users, doing their work without them might be considered a hassle or might not be possible to do at all. As the adage says, we only value things when we lose them. Hence, for determining whether a system is valuable for a group of users, a good exercise might be thinking of the consequences they might face if the system did not exist or stopped working.
- **Excitement**: Users might experience excitement when using computer systems. One example of such systems is computer games, which should be designed to generate excitement in the people who use them. Banking systems are not particularly seen as exciting. However, excitement can manifest in different ways, such as a person feeling astonished by how well a process has been automated, thus significantly easing the work to be done compared to manual operation. See *Characteristics of UX (desirable)* for an example.

- **Interest**: Systems that are interesting make users wonder how the built-in features work, encourage users to imagine new functionalities, or make users feel curious about whether these systems can communicate with other systems in order to improve operations across divisions. In summary, an interesting system makes users feel curious about exploring the different features the system has to offer.
- **Motivation**: Systems that improve the execution of processes motivate users to do their work. When systems make the execution of tasks tedious, users tend to feel demotivated.
- **Exclusivity**: When a system has been custom-designed to cover a division's specific needs, it will be seen as an exclusive system.
- **Impressiveness**: Users feel impressed when systems exceed their expectations (e.g., processes that used to take weeks are now executed in a matter of hours).

Definition of Appeal Polar Adjectives

These adjectives encompass a system's overall appeal for users.

- **Goodness**: This refers to whether a system is regarded as a good or bad product. As users, we tend to draw early conclusions from our first impression of a system (e.g., "That phone is so bad, I cannot even hold it properly" or "What an amazing car! Did you see those fine touches on the front bumper?").
- **Likeability**: This refers to whether users like systems, after being in touch with them. As users, we might think of a system as good or bad when we first interact with it. However, it is only when users interact with the system on a daily basis that they decide whether they really like the system or not.
- **Attractiveness**: This refers to how attracted users feel to systems after getting a sense of their ergonomic and hedonic qualities (e.g., accessibility, interface look and feel, clarity of information presented).
- **Friendliness**: After manipulating graphical interfaces, users might perceive them as friendly if the interfaces are easy to use. In contrast, users might perceive the interfaces as unfriendly if they are not intuitive or behave in an unpredictable manner.
- **Enjoyment**: Users might enjoy using systems that help them accomplish their

work smoothly. In contrast, they might feel annoyed when their work is hindered by systems that are poorly designed.

- **Aesthetics**: GUIs deliver an aesthetic impression, whether they were designed to be aesthetically pleasing or not. Many aspects impact aesthetics: the palette of colors used, the way information is grouped on the screen, the use of logos, icons, or animations for communicating affordances, etc.
- **Desirability**: That a system is desirable or not depends on whether it will help users achieve their goals more easily. A good exercise for getting an idea of how desirable a system might be is imagining all the things that would have to be done manually if the system did not exist and evaluating, based on that, whether users would advocate developing the system. If users would choose to have the system built, then the system could be regarded as desirable.

I will refer to this framework in the forthcoming chapters as the User Experience Framework (UXF).

Chapter 4. Benchmarking the Current State of UX in the Central Bank: A Survey Approach

Currently, the Bank does not have a way to measure the UX level of its computer systems. Furthermore, measuring the user experience is not a straightforward thing to do. First, it is necessary to rely on a framework that explains which are the UX dimensions (i.e., ergonomic quality, hedonic quality, and appeal) as well as their attributes (e.g., expectation, security, innovativeness, aesthetics). Second, it is important to suggest a method for measuring these attributes individually. Finally, a way to aggregate all individual results to determine the UX level has to be established.

The UXF suggests that the UX level of a system (i.e., its appeal) can be assessed directly by measuring its appeal attributes (e.g., goodness, friendliness, aesthetics), and indirectly by gauging its ergonomic (e.g., expectation, security, accessibility) and hedonic (e.g., innovation, excitement, motivation) attributes.

A way to capture users' appraisal for each UX attribute is a questionnaire based on a semantic differential scale (i.e., with polar adjectives on its extremes) [18], which would help in collecting users' appraisal for each attribute in all UXF dimensions (see Table 4). With the use of modern tools (e.g., Google Forms, Qualtrics, Survey Monkey), surveys can be applied to remote users (in this case the Bank's users) and their answers can be quickly analyzed to gain insights.

After some consideration, I decided to use the questionnaire approach with the semantic differential scale. A seven-point scale will be used to collect a user's opinion on each attribute. Five-point and six-point scales were also evaluated. The latter would prevent users from selecting a neutral stand, whereas the former would present users a smaller number of options to choose from. After evaluating some test users' opinions on the different scales, and in order to keep the right balance between providing users with a fair range of evaluation choices, a seven-point scale seemed to be the best choice. This scale is also meant to help lower the central tendency bias [18] factor.

Since users will have to choose a number between 1 and 7 for evaluating each UXF attribute (i.e., each pair of polar adjectives) it is important that they know the way every number will be interpreted by the survey's analyst. Figure 9 shows a proposal for the seven-point scale interpretation. On the one hand, the left side of the scale will hold the negative connotation of the attribute being evaluated, with numbers 1 to 3 loading towards the negative polar adjective. On the other hand, the right side of the scale will carry a positive connotation, with numbers 5 to 7 loading towards the positive adjective. Finally, number 4 will denote an equal balance between the negative and positive connotations.

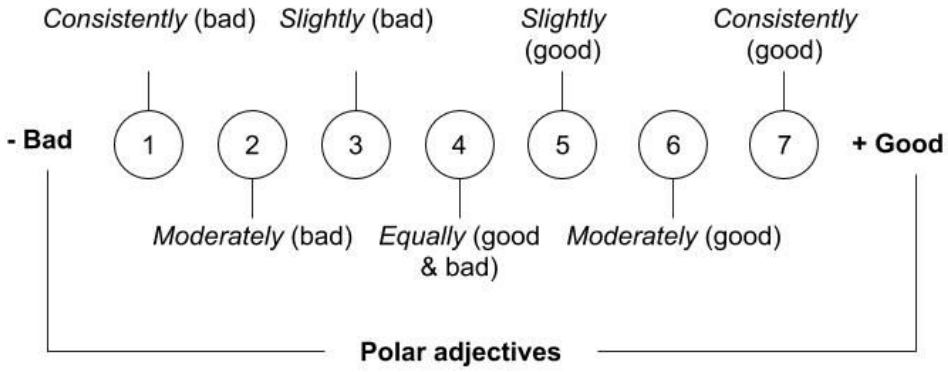


Figure 9. Seven-point scale interpretation.

Two surveys were made: one for systems' users and one for systems' developers. The idea behind two questionnaires is to compare the user's real UX appraisal of systems versus what developers think the user's UX appraisal is. All test users I asked agreed that contrasting both perspectives would be an enriching exercise.

The questionnaire for users will have eight sections:

1. **Landing page:** This explains the purpose of the survey, gives a brief definition of UX, shows the survey's authors, and the confidentiality policy.
2. **General Information:** This section asks the user for general information (i.e., their current division within the Bank, their seniority, and their academic background).

3. **Computer Systems Category Selection:** This section is crucial for the rest of the survey. Users are asked to select the category that represents most of the systems they use for doing their work. The Bank has many operating computer systems, most of them highly relevant for the daily operation. In order to evaluate their current UX level, users could answer a survey for each of the systems they are in touch with on a daily basis. However, this approach would demand considerably more time from users, and they might end up answering just for one system in most cases, thus decreasing the level of confidence per system.

   Another approach was to group systems into different categories according to the look and feel of their interfaces and the businesses they touch upon. This way, users evaluating the categories' UX attributes would also be evaluating all systems within that category at the same time. Since systems pertaining to one category have very similar graphical interfaces, users would be able to evaluate their UX attributes more easily as a whole. Given that more users would choose a category rather than a single system, the level of confidence per category would be higher than the one per system. For that reason, I decided to select the categories approach.

   After gathering the required information that would allow me to categorize all systems, I consulted an OSDD expert (a recently retired OSDD manager) who helped me do the grouping. After polishing the proposed grouping depending on how similar the systems' interfaces were among them, I came up with a fifteen-category list of categories that was included in the survey. For privacy reasons, I will not disclose any particular details about the systems included in the categories. Table 5 shows a list of all categories and their Graphical User Interface styles. Types A, B, or C refer to different front-end technologies. Even if some categories share more than one technology, I made sure that the systems within them had similar interfaces.

Table 5. Computer Systems Categories.

Abbreviated name	Related business and category name	Graphical User Interface style
DERA	Domestic Exchange Rates "A"	Standalone type A
DERB	Domestic Exchange Rates "B"	Web type A
DMAA	Domestic Markets Analysis "A"	Web type B
DMAB	Domestic Markets Analysis "B"	Standalone type B
FPA	Financial Programming "A"	Web type B
FPB	Financial Programming "B"	Standalone type B
IOMA	International Operations Management "A"	Standalone type C
IOMB	International Operations Management "B"	Standalone type A
MPFAA	Monetary Policy and Financial Agency "A"	Standalone type A
MPFAB	Monetary Policy and Financial Agency "B"	Standalone type D
MPFAC	Monetary Policy and Financial Agency "C"	Web type B
OMA	Operations Management "A"	Web type B
OMB	Operations Management "B"	Standalone type B
SOMA	Securities Operations Management "A"	Web type A
SOMB	Securities Operations Management "B"	Standalone type A

4. **Ergonomic Quality Part A:** This section asks users to evaluate the ergonomic quality of the category previously chosen. In order to do so, they are presented with the different ergonomic polar adjectives (see Table 4) and the seven-point scale to rate each of them based on the criteria shown in Figure 9.
5. **Ergonomic Quality Part B:** Since ergonomic quality is the dimension with most of the adjectives to evaluate, I decided to split it into two sections. This way, users would have the perception of making progress through the survey more quickly, making it easier to retain their attention.
6. **Hedonic Quality:** This section asks users to evaluate the hedonic quality of the category of systems previously chosen. As in the ergonomic section, they are presented with the different hedonic polar adjectives (see Table 4) and the seven-point scale to rate each of them based on the criteria shown in Figure 9.

7. **Appeal:** This section asks users to evaluate the appeal of the category previously chosen. In order to do so, they are presented with the different appeal polar adjectives (see Table 4) and the seven-point scale to rate each of them, based on the criteria shown in Figure 9. In the next chapter, I will use the UX index (a new construct based on the ergonomic and hedonic evaluations) to compare this appeal to the UX level, represented by the UX index. I am particularly interested in how appeal and UX level correlate through the UX index.
8. **Additional Comments:** here, users were asked to write down any additional comments they might have regarding their UX evaluation of the chosen category. This is an opportunity for them to directly express their feelings and concerns in a more personal fashion.

The survey for developers has the same structure, but they are asked to think in terms of the users' perspective rather than their own when answering questions about ergonomic and hedonic qualities. Additionally, in section 2, developers were asked how familiar they were with the term UX. The question about the division they work for was omitted for developers since that is already known.

The surveys' duration was determined by timing some test users. Answering all closed questions should not take more than ten minutes for users or developers. This duration seemed to be acceptable for keeping users' attention throughout the whole questionnaire. The survey's clarity in Spanish was also tested by the same test users, and some adjustments were made based on their comments in order to convey questions in a clearer and more concise fashion. Please refer to Appendices A and B for a detailed description of both surveys.

Measuring the UX level and Statistical Analysis

According to the UXF, a system's appeal is the result of a tradeoff between the system's ergonomic and hedonic qualities. The structure of the proposed survey for users will allow collection of the evaluation of UX dimensions: ergonomic quality, hedonic quality, and appeal, separately. Users will evaluate all attributes or nouns shown in Table 4. Ergonomic quality has fourteen variables or attributes. Hedonic quality has ten, and

appeal has seven. I will use ergonomic and hedonic variables (i.e., twenty-one in total) to determine the UX level (or indirect appeal) of OSDD systems. Then, I will compare this UX level to the measured appeal and determine whether they are correlated, as the UXF implies. To avoid confusion, I must clarify that I might use the terms UX level and UX index interchangeably. However, UX index will be mostly used when talking about statistical analysis.

In order to determine the UX index, I will run a factor analysis for ergonomic and hedonic variables (see Table 4) based on the survey results. This analysis will help me determine whether these variables measure a unidimensional construct. In case all variables load towards a single factor, I will then proceed to run a principal components analysis for determining the best variable coefficients to represent the UX index.

The UX level will be determined both directly by the appeal resulting from the survey, and indirectly by the UX index (i.e., factor analysis and principal components analysis of ergonomic and hedonic variables). In other words, it will not be necessary to determine the UX index in order to have a UX level measure, which will be clearly defined by the resulting survey's appeal. The only reason why the UX index will be calculated is to determine whether it is positively correlated with appeal. In case they are correlated, the UXF proposal regarding the UX level being the result of a trade-off between ergonomic and hedonic qualities would be supported.

For both surveys, I will present descriptive statistics that will show the overall evaluation of users and developers. Then, I will show more specific statistics by selected category, seniority, and background. Regarding the survey for developers, I will also show descriptive statistics based on how familiar they are with the term "UX". These statistics will allow me to spot UX opportunity areas, which I will discuss in Chapter 6. Additionally, I will contrast the opinions of users and developers regarding how systems are performing currently on each UX attribute. Additionally, I will contrast the opinion of users and developers by selected category.

Google Forms was used to create the surveys, distribute them, and collect answers from users and developers. The friendliness of this tool, along with the fact that it supports semantic differential scales, made it fit my survey needs seamlessly.

Chapter 5. Analysis of Survey Results

The surveys for users and developers remained opened for four weeks in order to account for Mexican holidays in between, as well as unforeseen delays. The most complicated part of applying the surveys was distributing them across all users and developers who were part of the surveys' target population. I decided to use a top-bottom approach for distributing the questionnaires across directors, managers, deputy managers, and analysts. This way, the top management would be aware that I was conducting the surveys and they would have a chance to look at them and make observations accordingly (e.g., privacy issues regarding whether sensitive information was being disclosed in the surveys, without me noticing it).

It turns out that the top-bottom distribution approach did not work as I was expecting when it came to distributing the questionnaires. Recent changes in the Bank due to political reasons increased the workload at all levels. This caused a two-week delay in the surveys' distribution, thus impacting my original schedule. I do not mention this as a complaint. On the contrary, I really appreciate the help the top management gave to me. I just want to point out how unforeseen events can hinder people's plans and that we should always plan accordingly.

In order to encourage people to answer the surveys, I ended up sending personal requests via email. This approach worked very well, and fortunately, I was able to collect most of the answers in just two weeks.

Regarding the survey for users, I was expecting to get around 80 answers based on the target population. I managed to collect 64 responses, which corresponds to around 80% of the potential respondents. Users evaluated 14 out of 15 categories of operating computer systems (see Table 5 for details on the categories). I will use this survey for determining a UX index based on the ergonomic and hedonic variables shown in Table 4. This index will hopefully allow me to support the hypothesis that appeal derives from a combination of hedonic and ergonomic qualities.

Regarding the survey for developers, I collected 28 responses out of 35 that I was expecting (80% response rate, the same as the survey for users). Developers evaluated only 11 categories, compared to 14 assessed by users. As I previously mentioned, this survey will allow me to contrast what developers think is the UX level of systems according to users versus what users actually think.

Descriptive Statistics per Category

Survey for Users

Figure 10 shows the percentage of respondents within categories and the number of responses per category from the survey (see Table 5 for category names). All categories obtained responses except DMAB (Domestic Markets Analysis "B"). FPA and MPFAB obtained 100% of participation rate. In contrast, MPFAC and DMAA only reached participation rates of 25% and 20%, respectively.

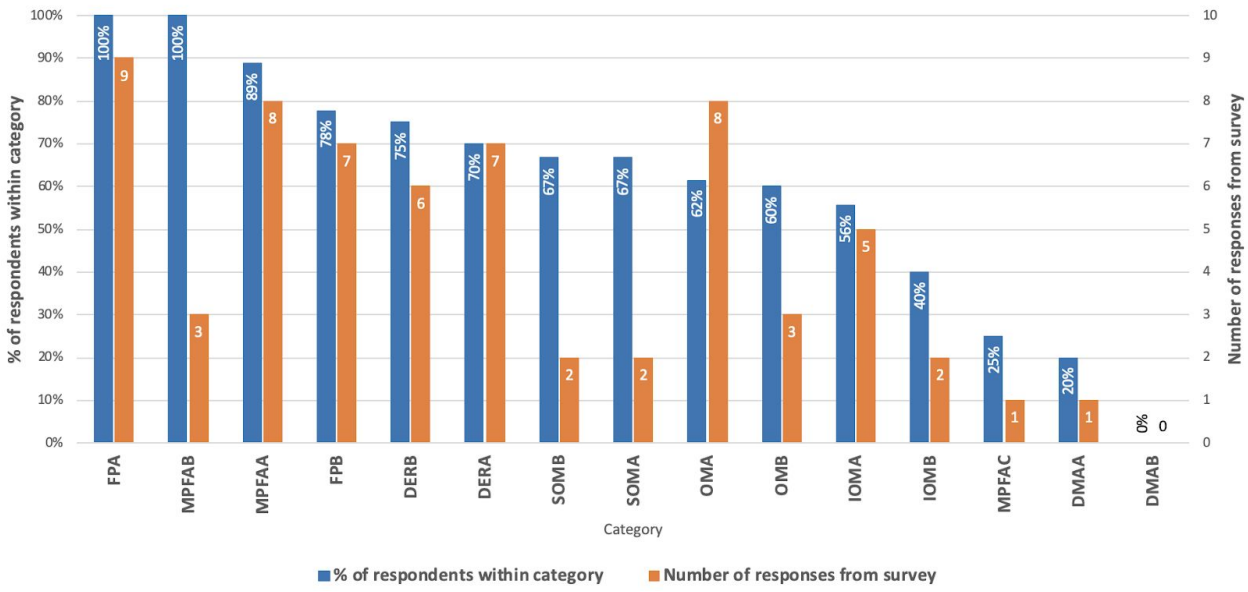


Figure 10. Percentage of Respondents vs Number of Responses, per Category.

Given the early stage of UX research, it is not clear which UX variables are the most important when it comes to generating an overall ergonomic, hedonic, and appeal scores. For that reason, all variables measured in the surveys (see Table 4) will be

considered to be equally important. Hence, ergonomic, hedonic, and appeal scores will be defined as the unweighted sum of all their respective variables.

It is important to be aware of the significance level that each score holds based on the percentage of respondents within each category. To analyze the survey for users, I will only consider categories whose percentage of respondents is higher than the first quartile (Q1 = 48%). Hence, categories IOBM, MPFAC, DMAA, and DMAB will be ignored. For each UX dimension (ergonomic, hedonic, and appeal), the analysis of categories will be based on the top (Q3) and bottom (Q1) quartiles of their mean scores.

Figure 11 shows the mean ergonomic quality score for all categories in descending order. The categories FPA (87.22), FPB (87.00), and MPFAA (83.63), scored above Q3 (81.40). In contrast, the categories OMA (73.50), SOMB (69.50), and IOMA (66.00), scored below Q1 (73.75). The maximum possible score for ergonomic quality is 98 (14 variables in a 7-point scale).

Next, I will do an analysis at the UX variable level, which will be based on Table 4 (The final list of polar adjectives classified by UX Dimension and Category), Figure 9 (Seven-point scale interpretation), and Table 9 (Mean Scores per Category across UX Variables). This analysis is intended to show what variables influenced ergonomic scores the most for the top (Q3) and bottom (Q1) categories. For simplifying the interpretation of the variables' mean scores, I will round them up if they have decimal values of 0.5 or higher, and round them down for decimal values lower than 0.5. For the categories that scored above Q3, I will analyze the UX variables that scored above Q3 (based on their scores' quartiles). Similarly, for the categories that scored below Q1, I will analyze the variables that scored below Q1 (based on their scores' quartiles).

Category FPA mean-scored above Q3 (6.33) for ergonomic variables: supportiveness (6.78), findability (6.78), and internal consistency (6.56). This can be interpreted as users perceiving systems within this category as *consistently* supportive, *consistently* findable, and *consistently* externally consistent. Category FPB mean-scored above Q3 (6.57) for ergonomic variable grouping (6.71). This can be interpreted as users perceiving systems within this category as *consistently* organized. Category MPFAA

mean-scored above Q3 (6.38) for ergonomic variable grouping (6.71). This can be interpreted as users perceiving systems within this category as *consistently* organized.

Category OMA mean-scored below Q1 (4.78) for ergonomic variables: efficiency (4.75), accessibility (4.63), simplicity (4.13), and external consistency (4.13). This can be interpreted as users perceiving systems within this category as *slightly* efficient, *slightly* accessible, *equally* complex and simple, and *equally* externally inconsistent and externally consistent. Category SOMB mean-scored below Q1 (4.00) for ergonomic variables: controllability (3.50), external consistency (3.00), and grouping (2.50). This can be interpreted as users perceiving systems within this category as *equally* uncontrollable and controllable, *slightly* externally inconsistent, and *slightly* cluttered. Category IOMA mean-scored below Q1 (4.05) for ergonomic variables: security (4.00), accessibility (3.60), learnability (3.40), and simplicity (3.00). This can be interpreted as users perceiving systems within this category as *equally* insecure and secure, *equally* inaccessible and accessible, *slightly* difficult to learn, and *slightly* complex.

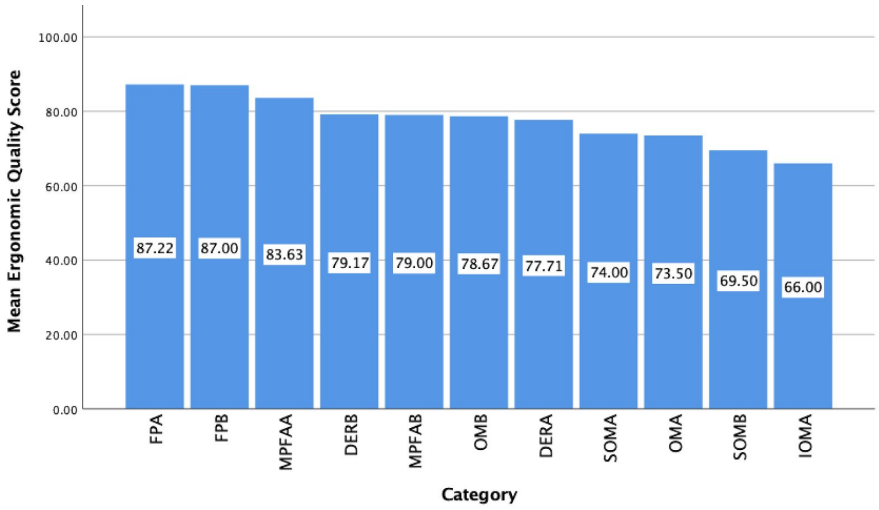


Figure 11. Mean Ergonomic Quality Score per category.

Figure 12 shows the mean hedonic quality score for all categories in descending order. The categories MPFAA (59.00), FPB (58.86), and FPA (87.22), scored above Q3 (58.28). In contrast, the categories OMA (49.75), DERA (47.86), and IOMA (44.60), scored below Q1 (50.63). The maximum possible score for hedonic quality is 70 (10 variables in a 7-point scale).

Category MPFAA mean-scored above Q3 (6.13) for hedonic variables: value (7.00) and exclusivity (6.50). This can be interpreted as users perceiving systems within this category as *consistently* valuable and *consistently* exclusive. Category FPB mean-scored above Q3 (6.11) for hedonic variables: value (6.71), exclusivity (6.29), and modernity (6.14). This can be interpreted as users perceiving systems within this category as *consistently* valuable, *moderately* exclusive, and *moderately* leading-edge.

In general, categories that scored high in ergonomic quality tended to score high in hedonic quality. This suggests that both qualities might be positively correlated.

Category OMA mean-scored below Q1 (4.66) for hedonic variables: impressiveness (4.63), creativeness (4.63), and excitement (4.00). This can be interpreted as users perceiving systems within this category as *slightly* impressive, *slightly* creative, and *equally* boring and exciting. Category DERA mean-scored below Q1 (4.14) for hedonic variables: modernity (4.00) and excitement (3.43). This can be interpreted as users perceiving systems within this category as *equally* usual and leading-edge, and *slightly* boring. Category IOMA mean-scored below Q1 (3.50) for hedonic variables: innovation (3.40), excitement (3.40), and modernity (3.00). This can be interpreted as users perceiving systems within this category as *slightly* conservative, *slightly* boring, and *slightly* usual.

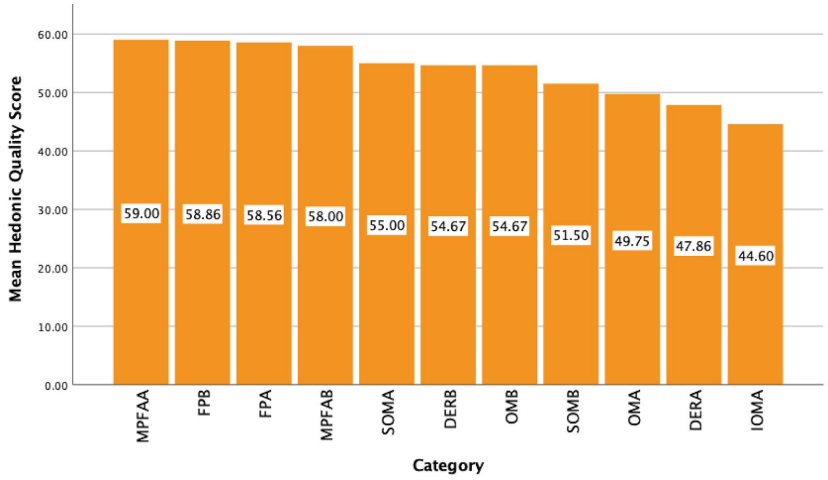


Figure 12. Mean Hedonic Quality Score per category.

Figure 13 shows the mean appeal score for all categories in descending order. The categories: FPB (42.71), SOMA (41.50), and MPFAA (40.75), scored above Q3 (40.54).

In contrast, categories: OMA (34.00), SOMB (34.00), and IOMA (30.60), scored below Q1 (35.14). The maximum possible score for appeal is 49 (7 variables in a 7-point scale). FPB scored first for appeal, and second for ergonomic and hedonic qualities. In contrast, IOMA scored last for all UX dimensions.

Category FBP mean-scored above Q3 (6.29) for appeal variables: desirability (6.71) and goodness (6.57). This can be interpreted as users perceiving systems within this category as being *consistently* desirable and *consistently* good as products. Category SOMA mean-scored above Q3 (6.25) for appeal variables: desirability (6.50), and friendliness (6.50). This can be interpreted as users perceiving systems within this category as being *consistently* desirable and *consistently* friendly. Category MPFAA mean-scored above Q3 (6.13) for appeal variables: desirability (7.00) and goodness (6.25). This can be interpreted as users perceiving systems within this category as being *consistently* desirable and *moderately* good as products.

Category OMA mean-scored below Q1 (4.56) for appeal variables: aesthetics (4.38) and attractiveness (4.13). This can be interpreted as users perceiving systems within this category as *equally* unaesthetic and aesthetic, and *equally* unattractive and attractive. Category SOMB mean-scored below Q1 (3.75) for appeal variables: attractiveness (3.00) and aesthetics (3.00). This can be interpreted as users perceiving systems within this category as *slightly* unattractive and *slightly* unaesthetic. Category IOMA mean-scored below Q1 (3.50) for appeal variables: aesthetics (2.80) and attractiveness (2.60). This can be interpreted as users perceiving systems within this category as *slightly* unaesthetic and *slightly* unattractive.

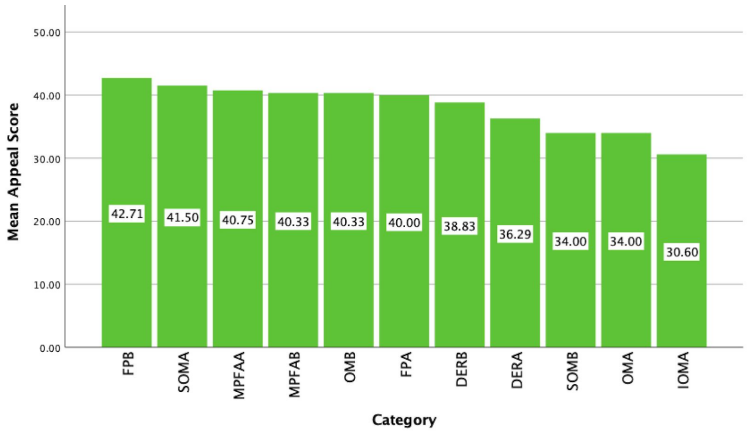


Figure 13. Mean Appeal Score per category.

The previous scores were the result of a simple summation of individual ergonomic, hedonic, and appeal variables. Consequently, it is not easy to tell how well the different categories did compared to the maximum possible scores. That is why I consider necessary to present percentage scores, and for Tables 6, 7, and 8 to show them for ergonomic, hedonic, and appeal, respectively. The interpretation of categories and their variables is the same as for mean scores. Table 9 shows the mean scores for UX variables across all categories.

Top (Q3) and bottom (Q1) quartiles for ergonomic percentage scores are 83.06% and 75.26%, respectively. For hedonic percentage scores, top (Q3) and bottom (Q1) quartiles are 83.25% and 72.32%, respectively. Q3 and Q1 for appeal are 82.74% and 71.72%, respectively.

Table 6. Percentage Ergonomic Quality Score.

	Percentage Mean Ergonomic Score						
Category	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
FPA	89.00%	11.20%	3.73%	93.88%	66.33%	100.00%	33.67%
FPB	88.78%	11.81%	4.46%	89.80%	65.31%	100.00%	34.69%
MPFAA	85.33%	9.56%	3.38%	87.76%	67.35%	94.90%	27.55%
DERB	80.78%	10.83%	4.42%	81.63%	64.29%	95.92%	31.63%
MPFAB	80.61%	5.40%	3.12%	82.65%	74.49%	84.69%	10.20%
OMB	80.27%	8.68%	5.01%	80.61%	71.43%	88.78%	17.35%
DERA	79.30%	11.67%	4.41%	81.63%	60.20%	95.92%	35.71%
SOMA	75.51%	18.76%	13.27%	75.51%	62.24%	88.78%	26.53%
OMA	75.00%	15.69%	5.55%	81.63%	45.92%	89.80%	43.88%
SOMB	70.92%	9.38%	6.63%	70.92%	64.29%	77.55%	13.27%
IOMA	67.35%	10.87%	4.86%	61.22%	57.14%	80.61%	23.47%

Table 7. Percentage Hedonic Quality Score.

	Percentage Mean Hedonic Score						
Category	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
MPFAA	84.29%	14.43%	5.10%	85.71%	52.86%	100.00%	47.14%
FPB	84.08%	16.00%	6.05%	88.57%	54.29%	100.00%	45.71%
FPA	83.65%	10.52%	3.51%	82.86%	70.00%	100.00%	30.00%
MPFAB	82.86%	5.15%	2.97%	84.29%	77.14%	87.14%	10.00%
SOMA	78.57%	10.10%	7.14%	78.57%	71.43%	85.71%	14.29%
DERB	78.10%	13.45%	5.49%	81.43%	52.86%	91.43%	38.57%
OMB	78.10%	5.41%	3.12%	75.71%	74.29%	84.29%	10.00%
SOMB	73.57%	9.09%	6.43%	73.57%	67.14%	80.00%	12.86%
OMA	71.07%	20.31%	7.18%	77.14%	30.00%	92.86%	62.86%
DERA	68.37%	14.60%	5.52%	70.00%	48.57%	91.43%	42.86%
IOMA	63.71%	14.80%	6.62%	60.00%	47.14%	85.71%	38.57%

Table 8. Percentage Appeal Score.

	Percentage Mean Appeal Score						
Category	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
FPB	87.17%	18.32%	6.92%	95.92%	51.02%	100.00%	48.98%
SOMA	84.69%	1.44%	1.02%	84.69%	83.67%	85.71%	2.04%
MPFAA	83.16%	18.28%	6.46%	89.80%	38.78%	93.88%	55.10%
OMB	82.31%	4.25%	2.45%	83.67%	77.55%	85.71%	8.16%
MPFAB	82.31%	4.25%	2.45%	83.67%	77.55%	85.71%	8.16%
FPA	81.63%	16.04%	5.35%	87.76%	57.14%	100.00%	42.86%
DERB	79.25%	11.65%	4.75%	81.63%	59.18%	91.84%	32.65%
DERA	74.05%	10.59%	4.00%	73.47%	59.18%	89.80%	30.61%
OMA	69.39%	25.30%	8.95%	76.53%	14.29%	89.80%	75.51%
SOMB	69.39%	5.77%	4.08%	69.39%	65.31%	73.47%	8.16%
IOMA	62.45%	14.04%	6.28%	61.22%	46.94%	79.59%	32.65%

Table 9. Mean Scores per Category across UX Variables for Users.
Keys: Orange (1 to 5), Yellow (>5), Blue (>6).

	Variable	DERA	DERB	FPA	FPB	IOMA	MPFAA	MPFAB	OMA	OMB	SOMA	SOMB
Appeal	Goodness	5.86	5.50	6.22	6.57	5.00	6.25	5.00	4.88	6.00	6.00	6.50
	Likeability	5.00	5.50	6.33	6.00	4.60	5.88	5.33	5.25	6.00	6.00	5.00
	Attractiveness	4.43	5.50	4.56	6.00	2.60	5.13	5.33	4.13	5.33	5.50	3.00
	Friendliness	5.43	5.67	6.22	6.00	4.60	6.00	6.00	5.25	6.00	6.50	5.00
	Enjoyment	4.71	5.17	5.67	5.71	4.20	5.25	5.33	4.75	5.67	5.50	4.50
	Aesthetics	4.43	5.33	4.22	5.71	2.80	5.25	6.33	4.38	5.67	5.50	3.00
	Desirability	6.43	6.17	6.78	6.71	6.80	7.00	7.00	5.38	5.67	6.50	7.00
Ergonomic Quality	Expectation	5.43	5.17	6.33	6.57	5.20	6.38	5.33	5.25	5.33	6.00	6.50
	Security	5.14	6.33	6.22	5.29	4.00	6.38	5.33	5.75	6.00	6.50	5.50
	Supportiveness	6.14	5.33	6.78	6.43	6.20	6.75	6.33	6.00	6.33	6.50	7.00
	Predictability	5.14	4.50	6.33	6.29	4.80	6.00	5.00	5.63	6.00	3.50	7.00
	Accessibility	5.29	5.33	6.00	5.86	3.60	4.50	2.33	4.63	4.67	5.00	5.00
	Findability	6.00	6.00	6.78	4.86	5.80	6.25	6.33	6.00	5.67	5.00	5.50
	Controllability	5.43	5.83	5.78	6.29	4.20	5.50	6.00	5.38	5.33	3.50	3.50
	Learnability	5.86	6.67	5.89	6.14	3.40	6.00	5.67	6.00	5.67	6.00	4.00
	Clarity	4.86	6.17	6.11	6.43	4.20	5.88	6.00	5.38	6.00	6.50	4.00
	Simplicity	5.14	4.83	6.00	6.57	3.00	5.50	6.67	4.13	5.33	5.00	4.50
	Internal Consistency	6.00	5.67	6.56	6.57	6.20	6.25	6.33	5.63	6.00	6.50	6.00
	External Consistency	5.57	6.00	5.78	6.43	5.60	5.25	6.00	4.13	5.33	3.50	3.00
	Efficiency	5.57	5.33	6.33	6.57	5.40	6.38	5.33	4.75	5.67	5.00	5.50
	Grouping	6.14	6.00	6.33	6.71	4.40	6.63	6.33	4.88	5.33	5.50	2.50
Hedonic Quality	Innovation	4.57	5.33	5.67	6.00	3.40	6.13	5.67	4.75	4.67	6.00	5.00
	Modernity	4.00	5.33	5.67	6.14	3.00	5.50	6.00	4.75	4.67	6.00	5.00
	Creativeness	4.14	5.17	5.78	6.00	3.80	5.88	6.00	4.63	4.33	5.00	4.50
	Originality	4.14	5.50	6.11	5.71	4.40	5.38	5.67	4.75	5.00	5.50	4.50
	Value	7.00	6.50	6.89	6.71	6.60	7.00	6.67	6.25	6.67	6.50	7.00
	Excitement	3.43	4.83	4.22	5.29	3.40	4.75	3.67	4.00	5.67	4.00	3.50
	Interest	5.14	5.50	5.89	5.71	5.80	6.13	6.00	5.38	5.67	5.50	4.50
	Motivation	5.43	5.50	5.78	5.71	4.80	6.13	6.33	5.50	5.67	5.50	5.50
	Exclusivity	5.57	6.00	6.67	6.29	5.20	6.50	6.67	5.13	7.00	6.00	7.00
	Impressiveness	4.43	5.00	5.89	5.29	4.20	5.63	5.33	4.63	5.33	5.00	5.00

Survey for Developers

Developers only responded for 11 out of 15 categories. Some categories obtained only 1 or 2 responses from developers (see Figure 14). This is not surprising and should not be taken as a lower significance level per category since there are many systems that are supervised only by one developer or two. Normally, during the development phase of a system, there are around five people involved (e.g., developers, testers, project leaders) without including users. After the system's rollout and stabilization, during a certain period that can go from a month to a year depending on the system's size, only one or two programmers take charge of its future maintenance. Most systems in the

OSDD are legacy systems, while only a small percentage represents new systems being developed. However, this small percentage tends to take most of the OSDD available resources.

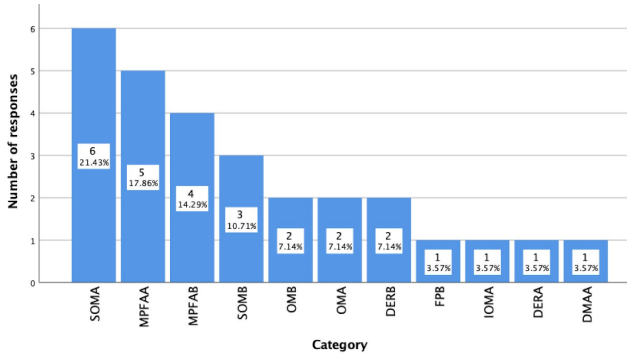


Figure 14. Number of responses per category.

Tables 10, 11, and 12 show more descriptive statistics for ergonomic, hedonic, and appeal, respectively. The mean scores are presented in descending order. Only four categories obtained more than 2 responses: MPFAA, MPFAB, SOMA, and SOMB. This was expected, as these categories either encompass more systems or a higher number of resources dedicated to maintaining the systems they include.

Top (Q3) and bottom (Q1) quartiles for ergonomic percentage scores are 79.85% and 76.79%, respectively. For hedonic percentage scores, top (Q3) and bottom (Q1) quartiles are 89.64% and 81.88%, respectively. Q3 and Q1 for appeal are 81.63% and 74.49%, respectively.

Table 10. Mean Percentage Ergonomic Quality Score for Developers.

	Percentage Ergonomic Score						
Category	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
DMAA	81.63%			81.63%	81.63%	81.63%	0.00%
MPFAA	81.63%	11.70%	5.23%	79.59%	67.35%	95.92%	28.57%
DERA	80.61%			80.61%	80.61%	80.61%	0.00%
SOMB	79.93%	4.12%	2.38%	77.55%	77.55%	84.69%	7.14%
DERB	79.59%	2.89%	2.04%	79.59%	77.55%	81.63%	4.08%
IOMA	78.57%			78.57%	78.57%	78.57%	0.00%
OMA	78.06%	6.49%	4.59%	78.06%	73.47%	82.65%	9.18%
OMB	78.06%	0.72%	0.51%	78.06%	77.55%	78.57%	1.02%
SOMA	76.36%	15.83%	6.46%	77.04%	53.06%	92.86%	39.80%
FPB	70.41%			70.41%	70.41%	70.41%	0.00%
MPFAB	68.11%	14.54%	7.27%	66.84%	52.04%	86.73%	34.69%

Table 11. Mean Percentage Hedonic Quality Score for Developers.

	Percentage Hedonic Score						
Category	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
IOMA	98.57%			98.57%	98.57%	98.57%	0.00%
SOMB	93.33%	0.82%	0.48%	92.86%	92.86%	94.29%	1.43%
MPFAA	90.00%	13.21%	5.91%	90.00%	74.29%	108.57%	34.29%
SOMA	88.57%	17.29%	7.06%	90.71%	61.43%	107.14%	45.71%
DERA	87.14%			87.14%	87.14%	87.14%	0.00%
DERB	85.00%	3.03%	2.14%	85.00%	82.86%	87.14%	4.29%
DMAA	82.86%			82.86%	82.86%	82.86%	0.00%
OMA	82.14%	7.07%	5.00%	82.14%	77.14%	87.14%	10.00%
MPFAB	81.79%	21.23%	10.62%	80.71%	57.14%	108.57%	51.43%
OMB	77.14%	6.06%	4.29%	77.14%	72.86%	81.43%	8.57%
FPB	71.43%			71.43%	71.43%	71.43%	0.00%

Table 12. Mean Percentage Appeal Score for Developers.

	Percentage Appeal Score						
Category	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
SOMB	85.03%	5.89%	3.40%	81.63%	81.63%	91.84%	10.20%
DMAA	83.67%			83.67%	83.67%	83.67%	0.00%
MPFAA	83.27%	13.72%	6.14%	85.71%	67.35%	100.00%	32.65%
DERB	81.63%	11.54%	8.16%	81.63%	73.47%	89.80%	16.33%
SOMA	81.63%	15.97%	6.52%	83.67%	57.14%	100.00%	42.86%
IOMA	81.63%			81.63%	81.63%	81.63%	0.00%
OMA	77.55%	8.66%	6.12%	77.55%	71.43%	83.67%	12.24%
FPB	77.55%			77.55%	77.55%	77.55%	0.00%
OMB	73.47%	2.89%	2.04%	73.47%	71.43%	75.51%	4.08%
MPFAB	72.45%	18.74%	9.37%	69.39%	53.06%	97.96%	44.90%
DERA	65.31%			65.31%	65.31%	65.31%	0.00%

Table 13. Mean Scores per Category across UX Variables for Developers.
Keys: Orange (1 to 5), Yellow (>5), Blue (>6).

	Variable	Category										
		DERA	DERB	DMAA	FPB	IOMA	MPFAA	MPFAB	OMA	OMB	SOMA	SOMB
Appeal	Goodness	6.00	6.50	6.00	4.00	6.00	6.20	6.00	6.00	5.50	6.50	7.00
	Likeability	5.00	6.00	5.00	5.00	6.00	5.80	5.00	5.50	5.00	5.67	6.33
	Attractiveness	3.00	5.50	5.00	5.00	6.00	5.00	5.25	4.50	5.00	5.33	5.33
	Friendliness	5.00	5.50	7.00	6.00	5.00	5.80	5.25	5.50	5.00	5.83	5.33
	Enjoyment	3.00	5.50	6.00	6.00	5.00	5.60	4.50	5.50	4.50	5.00	5.33
	Aesthetics	3.00	4.50	6.00	6.00	5.00	5.40	4.50	5.00	5.00	5.67	5.33
	Desirability	7.00	6.50	6.00	6.00	7.00	7.00	5.00	6.00	6.00	6.00	7.00
Ergonomic	Expectation	6.00	6.00	6.00	5.00	6.00	6.80	4.00	6.00	6.00	5.67	6.67
	Security	4.00	6.00	7.00	6.00	6.00	5.60	4.75	6.00	5.50	5.67	6.33
	Supportiveness	6.00	7.00	6.00	7.00	7.00	6.80	6.00	7.00	6.50	6.50	7.00
	Predictability	6.00	5.50	6.00	6.00	4.00	5.40	4.25	5.50	5.00	5.67	6.00
	Accessibility	2.00	3.00	5.00	2.00	5.00	3.80	3.00	3.00	4.00	3.33	3.67
	Findability	6.00	6.50	7.00	4.00	7.00	6.60	6.25	5.00	6.50	4.83	6.33
	Controllability	5.00	5.00	6.00	5.00	5.00	5.80	4.75	6.00	5.50	5.17	6.33
	Learnability	5.00	6.50	6.00	4.00	4.00	5.20	5.00	5.50	5.00	5.67	4.33
	Clarity	7.00	7.00	6.00	4.00	7.00	5.40	5.00	6.50	5.50	6.00	4.67
	Complexity	7.00	4.50	5.00	5.00	7.00	5.00	5.00	5.50	6.00	4.50	3.67
	Internal Consistency	7.00	6.00	5.00	7.00	7.00	6.40	5.00	5.50	6.00	5.50	6.00
	External Consistency	4.00	3.50	6.00	4.00	1.00	5.00	4.25	5.00	5.00	4.83	6.33
	Efficiency	7.00	6.00	6.00	6.00	6.00	5.80	4.50	5.50	5.50	5.67	6.33
	Grouping	7.00	5.50	3.00	4.00	5.00	6.40	5.00	4.50	4.50	5.83	4.67
Hedonic	Innovation	6.00	6.00	6.00	4.00	7.00	5.20	4.25	5.50	5.50	5.33	6.33
	Modernity	1.00	5.00	6.00	3.00	3.00	3.80	5.00	3.50	3.50	4.33	5.00
	Creativeness	5.00	4.50	6.00	4.00	7.00	5.20	5.25	4.50	4.50	5.17	5.00
	Originality	2.00	4.00	5.00	4.00	7.00	4.80	5.50	6.00	5.00	5.50	5.00
	Value	7.00	7.00	7.00	7.00	7.00	6.80	6.25	7.00	6.50	7.00	7.00
	Excitement	7.00	5.00	4.00	5.00	7.00	5.40	5.00	4.50	3.00	5.00	6.33
	Interest	7.00	5.00	5.00	5.00	7.00	5.20	5.00	5.50	4.50	5.67	6.67
	Motivation	7.00	5.50	5.00	4.00	5.00	6.60	4.75	5.50	4.50	5.67	6.33
	Exclusivity	7.00	6.50	6.00	5.00	7.00	7.00	6.00	6.00	6.50	7.00	6.67
	Impressiveness	5.00	5.50	5.00	5.00	7.00	6.60	5.25	5.00	6.00	5.50	6.33

Contrast between Surveys

Table 14 shows a comparison between the mean scores of users and developers across UX dimensions. Only the categories that obtained responses from both users and developers were included. The first and last rankings are bolded.

Table 14. Comparison of mean percentage scores across categories and UX dimensions between users and developers.

Category	Mean Percentage Score											
	Ergonomic				Hedonic				Appeal			
	Users	Ranking	Developers	Ranking	Users	Ranking	Developers	Ranking	Users	Ranking	Developers	Ranking
DERA	79.30%	6	80.61%	2	78.10%	5	87.14%	5	83.16%	3	72.45%	9
DERB	80.78%	3	79.59%	4	63.71%	**10**	82.14%	7	69.39%	8	77.55%	7
FPA	89.00%	**1**	70.41%	9	78.57%	4	88.57%	4	62.45%	**10**	81.63%	5
FPB	88.78%	2	78.57%	5	84.29%	**1**	85.00%	6	82.31%	4	81.63%	4
IOMA	67.35%	**10**	81.63%	**1**	68.37%	9	90.00%	3	87.17%	**1**	83.27%	2
MPFAB	80.61%	4	68.11%	**10**	73.57%	7	93.33%	2	79.25%	6	65.31%	**10**
OMA	75.00%	8	78.06%	6	82.86%	3	81.79%	8	82.31%	5	83.67%	**1**
OMB	80.27%	5	78.06%	7	71.07%	8	98.57%	**1**	74.05%	7	77.55%	6
SOMA	75.51%	7	76.36%	8	78.10%	6	71.43%	**10**	69.39%	9	73.47%	8
SOMB	70.92%	9	79.93%	3	84.08%	2	77.14%	9	84.69%	2	81.63%	3

Figure 15 shows the mean percentage ergonomic scores from users (blue) and developers (orange), as well as the delta between them (gray). The graph shows that in most cases, developers and users have aligned perceptions of what the ergonomic level of systems is. Significant differences (deltas) will be those above Q3 (11.93%) for delta values.

Based on the quartile criterion Q3 (11.93%), three categories present significant differences: FPA (18.59%), IOMA (14.29%), and MPFAB (12.50%). Categories FPA and MPFAB were scored higher by users. In contrast, category IOMA was scored higher by developers.

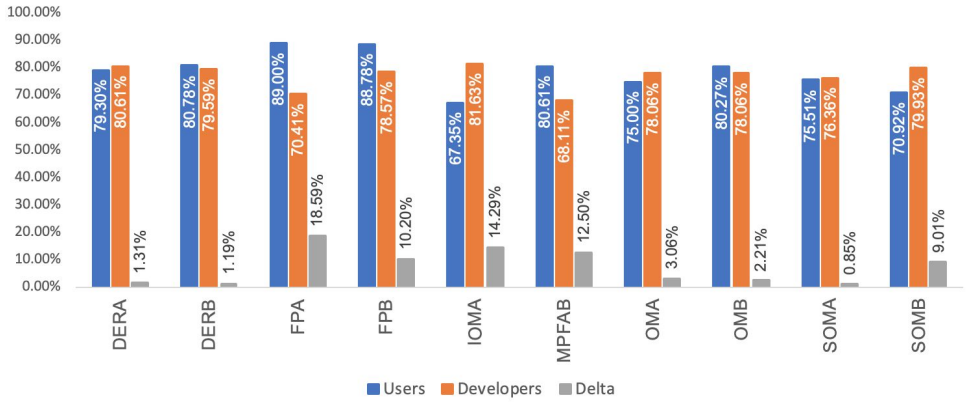


Figure 15. Comparison of Mean Percentage Ergonomic Scores between Users and Developers.

Figure 16 shows the mean percentage hedonic scores from users (blue) and developers (orange), as well as the delta between them (gray). Based on Q3 (19.43%) criterion for hedonic scores, three categories present significant differences: IOMA (21.63%), MPFAB (19.76%), and OMB (27.50%). The three categories were scored higher by developers.

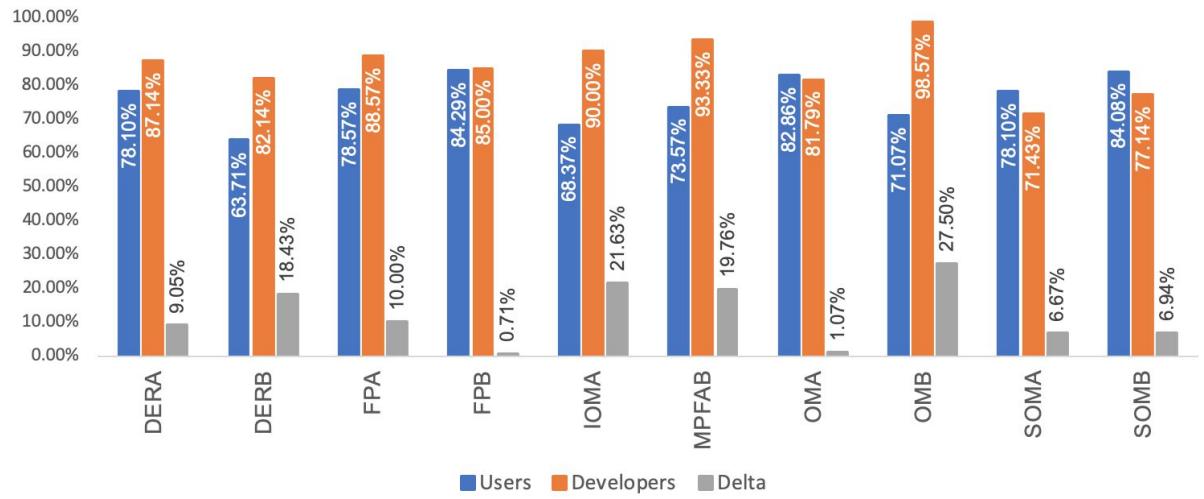


Figure 16. Comparison of Mean Percentage Hedonic Scores between Users and Developers.

Figure 17 shows the mean percentage appeal scores from users (blue) and developers (orange), as well as the delta between them (gray). Based on Q3 (11.93%) criterion for appeal scores, two categories present significant differences: FPA (19.18%) and MPFAB (13.94%). FPA was scored higher by users, whereas MPFAB was scored higher by developers.

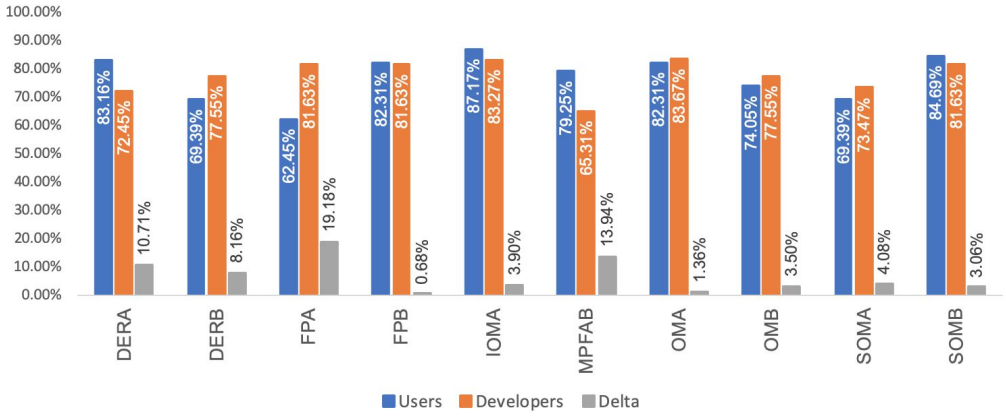


Figure 17. Comparison of Mean Percentage Appeal Scores between Users and Developers.

The analysis by category has shown that users perceive systems within categories FPA, FPB, and MPFAA, as having higher ergonomic quality. Categories FPA and FPA include systems that support the financial programming in the Bank. These systems scored higher for ergonomic variables: supportiveness, findability, internal consistency, and grouping. Systems within MPFAA category, support the monetary policy and financial agency businesses. These systems scored higher for ergonomic variable grouping. A common trait systems within these categories share, is that they are seen by users as consistently organized. The three categories

require organized interfaces since users of the systems there included do intensive data analysis and hold real-time operations with other institutions, which make the attention to detail very important.

On the other hand, users perceive systems within categories OMA, SOMB, and IOMA, as having lower ergonomic quality. Category OMA scored lower for ergonomic variables: efficiency, accessibility, simplicity, external consistency. Category SOMB scored lower for ergonomic variables: controllability, external consistency, and grouping. Category IOMA scored lower for ergonomic variables: security, accessibility, learnability, and simplicity. Categories OMA and SOMB share lower external consistency, and categories OMA and IOMA share lower accessibility.

Overall Descriptive Statistics

So far, the analysis has been done from a category perspective. I include this section in order to analyze the whole dataset at the UX variable level. This way, I will be able to see how users perceive individual UX attributes (variables) within the ergonomic, hedonic, and appeal dimensions.

Survey for Users

Figure 18 shows the mean scores for ergonomic variables (see Table 4) in descending order. The analysis will cover the top (Q3=5.81) and bottom (Q1=5.50) quartiles that represent the mean scores of ergonomic variables. In general, users think that operating systems are *moderately* supportive (6.31), *moderately* internally consistent (6.13), and *moderately* findable (5.94). In contrast, users think that systems are *slightly* controllable (5.47), *slightly* externally consistent (5.34), *slightly* simple (5.19), and *slightly* accessible (4.92).

There is a list of things developers focus their attention on when building a new system (e.g., expectation, security, usability), and I have never heard a developer mention accessibility at all. As engineers, we tend to assume everybody sees or hears the world the way we do, and we build accordingly. That is why it is not a surprise that accessibility is the variable with the lowest score. There is no doubt that computer systems should account for this if they are to be inclusive.

Ten out of 14 ergonomic variables mean-scored in the *moderately* zone, 4 scored *slightly*, and none scored in the *consistently* zone. This could be interpreted as users seeing operating systems as *moderately* ergonomic. Therefore, the OSDD may choose

to work in order to make systems that people can depend more on (e.g., more usable, more inclusive, less error-prone), that are perspicuous (e.g., easy to learn, easy to interpret, internally and externally consistent), and that have more efficient interfaces (e.g., the interface presents information in an organized fashion, interfaces allow users to work faster by avoiding redundant steps).

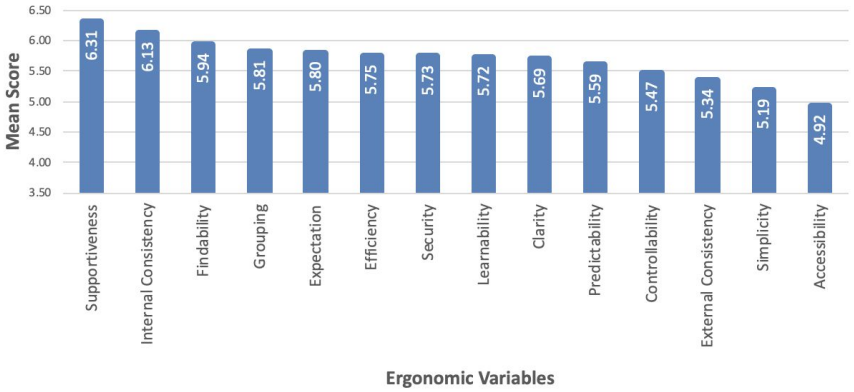


Figure 18. Mean Scores for Ergonomic Variables.

Figure 19 shows the mean scores for hedonic variables (see Table 4) in descending order. The analysis will cover the top (Q3=5.68) and bottom (Q1=5.10) quartiles that represent the mean scores of ergonomic variables. In general, users think that operating systems are *consistently* valuable (6.67), *moderately* exclusive (6.03), and *moderately* interesting (5.69). In contrast, users think that systems are *slightly* creative (5.09), *slightly* modern (5.05), and *equally* boring and exciting (4.30).

One hedonic variable mean-scored in the *consistently* zone, 3 scored *moderately*, 5 scored *slightly*, and 1 scored *equally*. Overall, users regard the OSDD systems as *slightly* good in terms of their hedonic quality. It is important to highlight that users perceive systems as valuable, which is reassuring since systems that do not deliver value are simply put away by users. The intended value of a system must be noticed before people engage in its development process. However, even if value is essential, it is not the only hedonic attribute a system should excel at in order to deliver superior hedonic quality. The OSDD might also care about building systems that are modern (i.e., systems that account for the latest interface trends), creative (i.e., do more with less), and even exciting (i.e., users see how a system considerably facilitates their everyday activities at work).

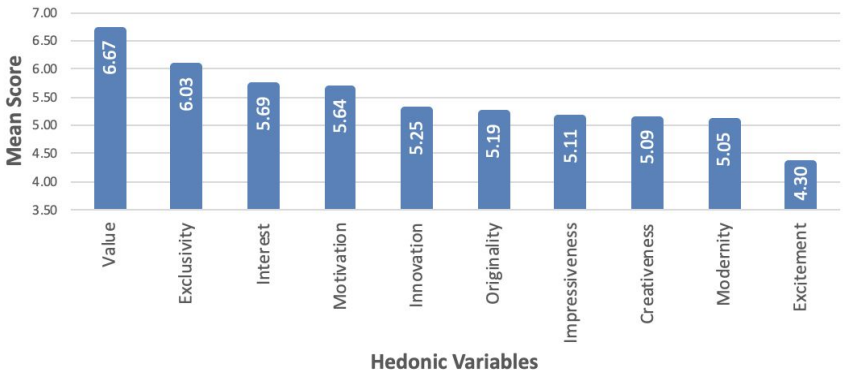


Figure 19. Mean Scores for Hedonic Variables.

Figure 20 shows the mean scores for appeal variables (see Table 4) in descending order. The analysis will cover the top (Q3=5.77) and bottom (Q1=5.02) quartiles that represent the mean scores of ergonomic variables. In general, users think that operating systems are *moderately* desirable (although they are very close to *consistently* desirable), and *moderately* good as products. In contrast, users think that operating systems are *slightly* aesthetic, and *slightly* attractive. OSDD systems have a high level of desirability for users, as the negative impact on their work would be considerably high if they did not exist. This is aligned with how valuable users think OSDD systems are for them. In contrast, the OSDD might want to improve its systems' aesthetics and attractiveness.

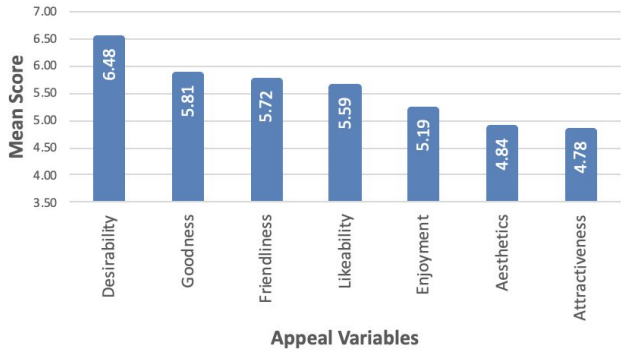


Figure 20. Mean Scores for Appeal Variables.

Table 15 shows descriptive statistics for all variables. Most of them mean-scored between 5 and 6, thus loading towards the positive connotation in the 7-point scale (see Figure 9). On the ergonomic side, supportiveness obtained the higher mean score (6.31), while accessibility the lowest (4.92). For hedonic variables, value was what users perceived as the best attribute of systems (6.67), while excitement was the lowest

(4.30). Regarding appeal, variables desirability and attractiveness represented the highest (6.48) and lowest (4.78) mean scores, respectively.

At the UX dimension level, ergonomic quality, hedonic quality, and appeal mean-scored 5.67, 5.40, and 5.49, respectively (see Table 16). It is interesting that the mean between ergonomic and hedonic qualities (5.54) is pretty close to the appeal mean score (5.49). The appeal mean score could be interpreted as users perceiving the UX level of OSDD systems as *slightly* positive (see Figure 9), thus leaving room for improvement.

Table 15. Descriptive Statistics per UX Variable.
Keys: Orange (1 to 5), Yellow (>5), Blue (>6).

	Variable	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
Appeal	Goodness	5.81	1.18	0.15	6	1	7	6
	Likeability	5.59	1.24	0.16	6	1	7	6
	Attractiveness	4.78	1.74	0.22	5	1	7	6
	Friendliness	5.72	1.27	0.16	6	1	7	6
	Enjoyment	5.19	1.28	0.16	5	1	7	6
	Aesthetics	4.84	1.85	0.23	5	1	7	6
	Desirability	6.48	1.04	0.13	7	1	7	6
Ergonomic Quality	Expectation	5.80	1.14	0.14	6	3	7	4
	Security	5.73	1.29	0.16	6	2	7	5
	Supportiveness	6.31	0.94	0.12	7	3	7	4
	Predictability	5.59	1.39	0.17	6	1	7	6
	Accessibility	4.92	1.87	0.23	5	1	7	6
	Findability	5.94	1.38	0.17	6	1	7	6
	Controllability	5.47	1.53	0.19	6	1	7	6
	Learnability	5.72	1.27	0.16	6	1	7	6
	Clarity	5.69	1.28	0.16	6	2	7	5
	Simplicity	5.19	1.70	0.21	6	1	7	6
	Internal Consistency	6.13	1.00	0.13	6	3	7	4
	External Consistency	5.34	1.55	0.19	6	1	7	6
	Efficiency	5.75	1.10	0.14	6	2	7	5
	Grouping	5.81	1.36	0.17	6	2	7	5
Hedonic Quality	Innovation	5.25	1.45	0.18	6	1	7	6
	Modernity	5.05	1.68	0.21	6	1	7	6
	Creativeness	5.09	1.43	0.18	6	1	7	6
	Originality	5.19	1.59	0.20	6	1	7	6
	Value	6.67	0.67	0.08	7	4	7	3
	Excitement	4.30	1.76	0.22	5	1	7	6
	Interest	5.69	1.14	0.14	6	2	7	5
	Motivation	5.64	1.34	0.17	6	1	7	6
	Exclusivity	6.03	1.25	0.16	7	2	7	5
	Impressiveness	5.11	1.50	0.19	5	1	7	6

Table 16. Mean Scores across UX Dimensions.

UX Dimension	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
Appeal	5.49	0.60	0.23	5.59	4.78	6.48	1.70
Ergonomic Quality	5.67	0.36	0.10	5.73	4.92	6.31	1.39
Hedonic Quality	5.40	0.64	0.20	5.22	4.30	6.67	2.38

Tables 17, 18, and 19 show the percentage mean scores from the division, seniority, and background angles. The column count represents the number of users that answered the survey for a specific division, seniority, or background.

On the division side, the Directorate of Domestic Operations rated the OSDD systems with the highest percentage scores for all UX dimensions, being the hedonic quality the lowest one. In contrast, the Directorate of International Operations rated OSDD systems with the lowest percentage scores, being the hedonic quality the lowest one, once again (only one response; lower significance level). The Directorate of Operations Support took a middle stance in all UX dimensions, being hedonic quality the one with the lowest score, in alignment with the other divisions.

Table 17. Percentage Mean Scores across UX Dimensions by Division.

UX Dimension	Division					
	Directorate of Operations Support		Directorate of International Operations		Directorate of Domestic Operations	
	Mean	Count	Mean	Count	Mean	Count
Ergonomic Quality	80.56%	38	61.22%	1	82.49%	25
Hedonic Quality	76.69%	38	55.71%	1	78.74%	25
Appeal	76.96%	38	61.22%	1	81.31%	25

Regarding seniority, users from 0 to 10 years gave the OSDD systems the highest percentage score for appeal (78.48%), followed by users with more than 10 years and up to 20 (78.23%). Finally, users with more than 20 years and up to 30, and users with more than 30 years rated OSDD systems with the lowest percentage mean appeal (77.55%). In general, the younger the user, the higher the appeal of systems.

Table 18. Percentage Mean Scores across UX Dimensions by Seniority.

UX Dimension	Seniority							
	0 to 10 years		More than 10 and up to 20 years		More than 20 and up to 30 years		More than 30 years	
	Mean	Count	Mean	Count	Mean	Count	Mean	Count
Ergonomic Quality	80.79%	52	81.18%	9	86.22%	2	80.61%	1
Hedonic Quality	77.01%	52	75.56%	9	90.00%	2	74.29%	1
Appeal	78.49%	52	78.23%	9	77.55%	2	77.55%	1

Regarding background, users in actuary gave OSDD systems the higher percentage score for appeal, while users in informatics gave systems the lowest appeal. Computer Systems or related was left out since it obtained only one response, and it represented an outlier for the analysis by background. In general, financial-related backgrounds tended to give higher appeal scores.

Table 19. Percentage Mean Scores across UX Dimensions by Background.

UX Dimension	Background											
	Actuary		Accounting		Economy		Finance		Informatics		Other	
	Mean	Count	Mean	Count	Mean	Count	Mean	Count	Mean	Count	Mean	Count
Ergonomic Quality	86.46%	11	78.43%	7	83.86%	32	86.73%	3	64.46%	6	71.68%	4
Hedonic Quality	85.71%	11	76.53%	7	78.44%	32	82.38%	3	58.33%	6	75.00%	4
Appeal	87.38%	11	75.80%	7	80.23%	32	80.95%	3	58.50%	6	81.63%	4

Tables 20, 21, and 22 show the mean scores from the division, seniority, and background angles. The interpretation is the same as previous tables, but the means are based on the 7-point scale (see Figure 9), which uses adverbs *equally*, *slightly*, *moderately*, and *consistently* for interpreting scores with either negative or positive connotations.

Table 20. Mean Scores across UX Dimensions by Division.

UX Dimension	Division					
	Directorate of Operations Support		Directorate of International Operations		Directorate of Domestic Operations	
	Mean	Count	Mean	Count	Mean	Count
Ergonomic Quality	5.64	38	4.29	1	5.77	25
Hedonic Quality	5.37	38	3.90	1	5.51	25
Appeal	5.39	38	4.29	1	5.69	25

Table 21. Mean Scores across UX Dimensions by Seniority.

UX Dimension	Seniority							
	0 to 10 years		More than 10 and up to 20 years		More than 20 and up to 30 years		More than 30 years	
	Mean	Count	Mean	Count	Mean	Count	Mean	Count
Ergonomic Quality	5.66	52	5.68	9	6.04	2	5.64	1
Hedonic Quality	5.39	52	5.29	9	6.30	2	5.20	1
Appeal	5.49	52	5.48	9	5.43	2	5.43	1

Table 22. Mean Scores across UX Dimensions by Background.

UX Dimension	Background											
	Actuary		Accounting		Economy		Finance		Informatics		Other	
	Mean	Count	Mean	Count	Mean	Count	Mean	Count	Mean	Count	Mean	Count
Ergonomic Quality	6.05	11	5.49	7	5.87	32	6.07	3	4.51	6	5.02	4
Hedonic Quality	6.00	11	5.36	7	5.49	32	5.77	3	4.08	6	5.25	4
Appeal	6.12	11	5.31	7	5.62	32	5.67	3	4.10	6	5.71	4

Survey for Developers

Tables 23, 24, and 25 show descriptive statistics for ergonomic quality, hedonic quality, and ergonomic quality, according to developers. Variables are shown in descending order according to their mean scores. Figure 21 shows the overall mean scores across UX dimensions.

Top (Q3) and bottom (Q1) quartiles for ergonomic scores are 5.79 and 5.24, respectively. For hedonic scores, top (Q3) and bottom (Q1) quartiles are 5.71 and 5.09, respectively. Q3 and Q1 for appeal are 5.88 and 5.13, respectively.

Table 23. Descriptive Statistics for Ergonomic Variables.
Keys: Orange (1 to 5), Yellow (>5), Blue (>6).

Variable	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
Supportiveness	6.61	0.63	0.12	7.00	5.00	7.00	2.00
Findability	5.93	1.27	0.24	6.00	2.00	7.00	5.00
Internal Consistency	5.86	1.15	0.22	6.00	3.00	7.00	4.00
Expectation	5.82	1.16	0.22	6.00	2.00	7.00	5.00
Clarity	5.68	1.36	0.26	6.00	2.00	7.00	5.00
Efficiency	5.68	1.06	0.20	6.00	3.00	7.00	4.00
Security	5.64	1.03	0.19	6.00	3.00	7.00	4.00
Controllability	5.43	1.07	0.20	6.00	3.00	7.00	4.00
Predictability	5.36	1.19	0.23	6.00	3.00	7.00	4.00
Grouping	5.32	1.36	0.26	5.50	3.00	7.00	4.00
Learnability	5.21	1.26	0.24	5.50	2.00	7.00	5.00
Complexity	4.96	1.73	0.33	5.00	1.00	7.00	6.00
External Consistency	4.71	1.38	0.26	4.00	1.00	7.00	6.00
Accessibility	3.43	1.43	0.27	3.00	1.00	6.00	5.00

Table 24. Descriptive Statistics for Hedonic Variables.
Keys: Orange (1 to 5), Yellow (>5), Blue (>6).

Variable	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
Value	6.82	0.48	0.09	7.00	5.00	7.00	2.00
Exclusivity	6.57	0.74	0.14	7.00	5.00	7.00	2.00
Impressiveness	5.75	1.24	0.23	6.00	3.00	7.00	4.00
Motivation	5.61	1.20	0.23	5.00	4.00	7.00	3.00
Interest	5.50	1.32	0.25	5.00	2.00	7.00	5.00
Innovation	5.39	1.59	0.30	6.00	1.00	7.00	6.00
Excitement	5.14	1.58	0.30	5.50	1.00	7.00	6.00
Creativeness	5.07	1.25	0.24	5.00	2.00	7.00	5.00
Originality	5.07	1.46	0.28	5.00	2.00	7.00	5.00
Modernity	4.18	1.59	0.30	4.00	1.00	7.00	6.00

Table 25. Descriptive Statistics for Appeal Variables.
Keys: Orange (1 to 5), Yellow (>5), Blue (>6).

Variable	Mean	Standard Deviation	Standard Error of Mean	Median	Minimum	Maximum	Range
Desirability	6.25	1.43	0.27	7.00	2.00	7.00	5.00
Goodness	6.18	0.77	0.15	6.00	4.00	7.00	3.00
Friendliness	5.57	0.88	0.17	5.00	4.00	7.00	3.00
Likeability	5.57	1.00	0.19	6.00	4.00	7.00	3.00
Aesthetics	5.14	1.11	0.21	5.00	3.00	7.00	4.00
Enjoyment	5.11	1.34	0.25	5.00	3.00	7.00	4.00
Attractiveness	5.11	1.07	0.20	5.00	3.00	7.00	4.00

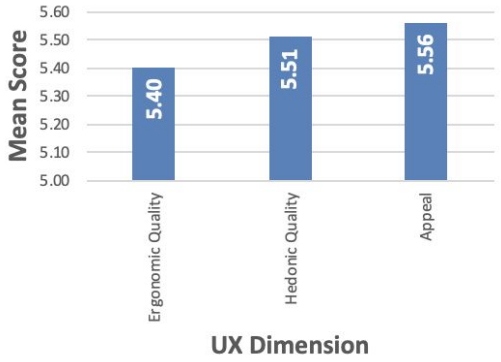


Figure 21. Mean Scores across UX Dimensions.

Contrast between Surveys

Figure 22 shows the mean scores for ergonomic variables from users (blue) and developers (orange), as well as the delta between them (gray). The graph shows that in most cases, developers and users have similar perceptions of the ergonomic level of systems. I will focus on the top quartile (Q3=0.44) of the deltas for analyzing the differences between users and developers for ergonomic variables.

Based on the previous criterion, accessibility (1.49), external consistency (0.63), grouping (0.49), and learnability (0.50) would be the ergonomic variables that present the highest differences between users and developers. Even though users perceive the accessibility level of OSDD systems higher than developers do, scores from both are in the *slightly* positive and *slightly* negative scale, respectively. Therefore, developers may

want to work to push this characteristic to the *moderately* positive or even the *consistently* positive zones (see Figure 9). External consistency, grouping, and learnability were also scored higher by users.

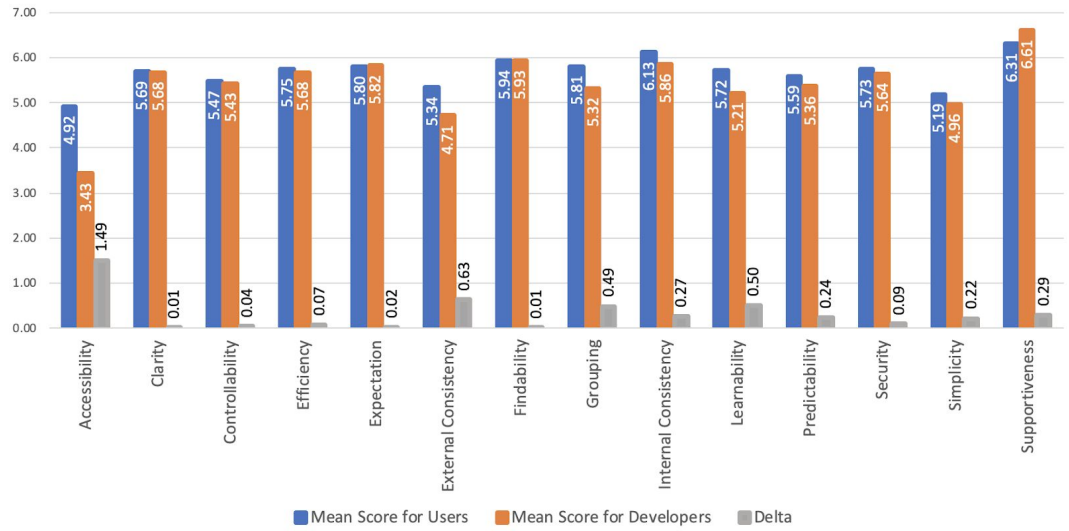


Figure 22. Mean Scores for Ergonomic Variables.

Figure 23 shows the mean scores for hedonic variables from users (blue) and developers (orange), as well as the delta between them (gray). I will focus on the top quartile (Q3=0.62) of the deltas for analyzing the differences between users and developers for hedonic variables. Based on the previous criterion, excitement (0.85), impressiveness (0.64), and modernity (0.87), present the highest differences between users and developers. Only modernity was scored higher by users.

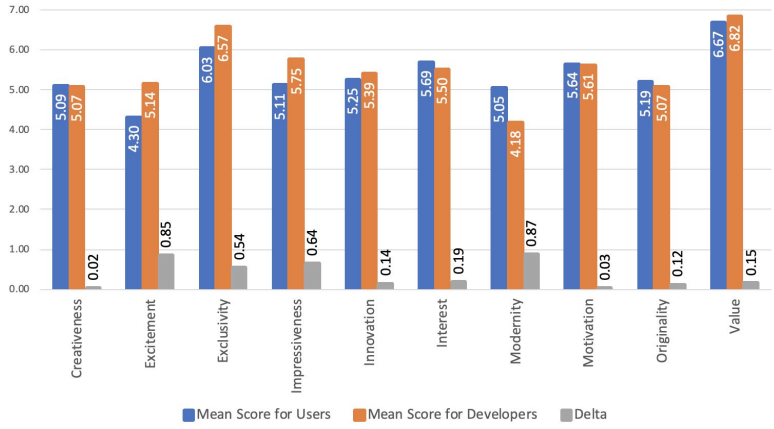


Figure 23. Mean Scores for Hedonic Variables.

Figure 24 shows the mean scores for appeal variables from users (blue) and developers (orange), as well as the delta between them (gray). I will focus on the top quartile (Q3=0.42) of the deltas for analyzing the differences between users and developers for appeal variables. Based on the previous criterion, enjoyment (0.70) and goodness (0.46) present the highest differences between users and developers. Enjoyment was scored higher by users, whereas goodness was scored higher by developers. Overall, users and developers have similar perspectives regarding the directly measured UX level (appeal) of OSDD systems.

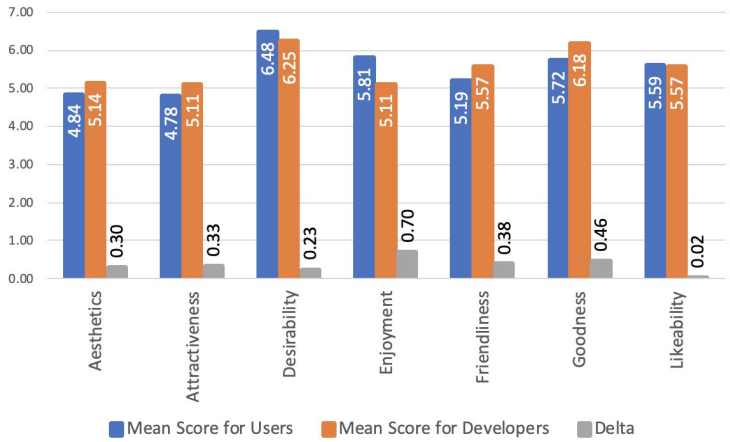


Figure 24. Mean Scores for Appeal Variables.

In summary, no ergonomic variable scored in the *consistently* zone. The highest scored ergonomic variables were supportiveness, internal consistency, and findability, whereas the lowest scored were controllability, external consistency, simplicity, and accessibility.

Only one hedonic variable scored in the consistently zone (value). Exclusivity and interest were scored in the *moderately* zone. Variables creative, modernity, and excitement, loaded towards the *slightly* zone.

Aesthetics and attractiveness were the less appealing characteristics of systems. Appeal variables desirability and goodness scored in the *moderately* zone.

Open Comments

Survey for Users

Thirteen comments were provided by users originally. I classified them depending on the UX attributes they were most related, as well as in "other items" (shown in *italic*) for those comments that did not fit into a UX attribute. Some comments touched upon UX attributes and "other items" at the same time, therefore I decided to split them into two comments. Other comments were related to more than one UX attribute, but it did not make sense to split them since their context would have been lost. Hence, I decided to indicate all UX attributes the comments were related to. I marked full comments or sections of them with + and - signs to indicate a positive (users praise systems) or negative connotation (users complain about the system, it seems users resist change, etc.). In total, sixteen comments were identified (some derived from the original thirteen) and Table 26 presents their arrangement.

Table 26. Open Comments from Users.
Keys: (+) positive connotation, (-) negative connotation.

UX Attributes or other Items	Comment
Modernity	(-) Systems have to provide tools for data analysis, especially for data visualization.
Predictability	(-) Errors in calculations have become recurrent.
Expectation	(-) Systems are not updated and generate errors that are given low priority by the OSDD when it comes to addressing them, thus leaving them to cumulate.
	(-) Systems present errors due to a lack of integral solutions. Since the general context of the system is not taken into account when developing or changing functionalities, solutions partially meet expectations. Sometimes, superficial solutions lead to unexpected behaviors (e.g., user profile is not taken into account, complements do not activate when they should).
	(-) Some users' requests have not been addressed by the OSDD.
External consistency	(-) Some OSDD systems have different affordances for executing the same task (e.g., different icons), thus increasing their systems' learning curves.
Interest	(-) Beta tests consume a great deal of users' time due to a non-tested interface[13] connectivity among systems. That is, tests are executed slowly due to communication errors among systems. Therefore, it is important to know how systems communicate with

[13] In this context, interface refers to the data structures interchanged by systems when they communicate and not to graphical user interfaces.

	each other and test their communication interfaces in order to make beta testing smoother.
Goodness	(+) In general, OSDD systems are seen as good products.
	(+) System B (an update of system A to a new technology; included in Monetary Policy and Financial Agency "A") is a great and promising system, as the current ones are too (the ones that have not been migrated to the new technology).
Clarity and Friendliness	(-) Many functionalities have been added to System C, thus making it look saturated and hard to operate.
Simplicity and Interest	(-) Is not easy to understand how systems work, thus making it hard to propose new functionalities.
Aesthetics and Accessibility	(+) Systems aid users do their work. (-) However, aesthetics and accessibility should be improved.
Supportiveness, Value, and Desirability	(+) System D ease users' daily activities at work and it is indispensable to apply surveys to other institutions.
Survey structure	(-) Survey should give respondents the opportunity to add custom comments for every question.
User engagement	(- not done currently/+ a good thing to be done) Users should engage in the modernization and efficiency of OSDD systems.
	(- not done currently/+ a good thing to be done) Users should be taken into account when developing interfaces, since they have experience regarding how interfaces work in other banks.

Survey for Developers

Thirteen comments were provided by developers originally. I classified them depending on the UX attributes they were related to the most, as well as in "other items" (shown in *italic*) for those comments that did not fit into a UX attribute. Some comments touched upon UX attributes and "other items" at the same time. Therefore, I decided to split them into two comments. Other comments were related to more than one UX attribute, but it did not make sense to split them since their context would have been lost. Hence, I decided to indicate all UX attributes the comments were related to. I considered that one comment (in *italics*) could be classified either as Goodness or Users' resistance to change; therefore, I put it in both. In total, fifteen comments were identified (some derived from the original thirteen) and Table 27 presents their arrangement.

Table 27. Open Comments from Developers.
Keys: (+) positive connotation, (-) negative connotation.

UX Attributes or other Items	Comment
Modernity	(-) OSDD systems have interfaces that have not been updated.
	(+) System E continues to be remarkable, even thought it was built ten years ago. (-) However, it has many aspects that should be modernized.
Expectation	(+) As developers, we focus on providing the requested functionality, (-) but care little about the user experience.
	(-) Systems present some issues when it comes to exporting information.
Exclusivity, Friendliness, and Supportiveness	(+) Most systems built in the OSDD are custom-made. (-) Users normally care about the automation of their processes rather than the interface. However, I believe interfaces are very important since they are tools users utilize every day at work. (+) That is why I have tried to deliver interfaces that look clean, integrate all functionalities requested by users, and that make the life of users easier.
Aesthetics and Value	(+) Users are used to do one or three clicks at most to get most of their processes done. (-) There are old systems, made with old technologies whose migration to new technologies has been really hard because they have considerably grown in functionalities. That is why these systems' interfaces are not that good, but these systems' value resides in the businesses they handle in the Bank.
Goodness	(+) Systems have very good interfaces.
	(+) Users are satisfied with systems built by the OSDD.
	(+) *I consider that users are satisfied with systems under category MPFAA due to the negative reaction they had when the OSDD proposed to switch to a new technology that provides similar mechanisms for supporting the daily operation.*
User engagement	(+) Users' opinions are valuable, (-) but we (developers) consider that they will not have time to devote to the design of interfaces.
	(+) Users are excited about requesting graphical components that are innovative, especially graphs that allow decision making in matter of seconds. Therefore, the interaction of users has become crucial for analyzing and designing these controls along with developers. The goal is to design controls that allow systems to report updates that reflect the operation in real-time.
	(-) I consider that users get involved very little in the design of interfaces because either nobody forces them to do it or nobody invites them to do it. There is no initiative from the top management to make users understand that systems are custom-made, and they should be involved in their construction from the beginning. The OSDD knows that users are very busy, and that is why it does not invite them to participate in the interface development. Consequently, users do not get involved and end up complaining about the OSDD making all decisions by

	arguing they did not ask for a certain functionality. Users have to get involved if the UX is to be improved.
Users' resistance to change	(-) *I consider that users are satisfied with systems under category MPFAA due to the negative reaction they had when the OSDD proposed to switch to a new technology that provides similar mechanisms for supporting the daily operation* (same comment as in goodness).
	(-) The construction of system F was a lesson in terms of users' resistance to change, since the OSDD tried to substitute an old system to which users were accustomed. (+) UX techniques can be of help in cases like this.
	Users were accustomed to system E. (-) Consequently, many changes had to be done to its updated version system F in order to make it look alike to the previous one.
Clarification	Answers were given for system G in particular (DERA category).

Contrast between Open Comments

Some comments from users reflect that systems fail to fulfill user expectations (do-goals), especially when it comes to addressing production errors. Users think that the OSDD does not address these kinds of errors in a timely manner, causing these to cumulate, which can become a burden for users. On the other hand, developers think that their systems fulfill user expectations, and only fail to account for the user experience in general.

Users and developers agree that there is room for improvement when it comes to the system's aesthetics. Users feel systems' accessibility should be improved, while developers did not comment about it. Users and developers see modernity as something systems lack. Users and developers think that users should get more involved in the interface design process.

Users and developers agreed that OSDD systems bring value to users' everyday activities at work.

Finally, developers think users tend to resist change, thus preferring to keep working with current systems as they got used to them after many years of interaction. Developers mentioned that most of the times, when a system gets finally updated, users request that most features from the previous version are also present in the new one, even erroneous behavior the old systems presented.

Determining the UX Index

The analyses done so far have focused either on the mean scores or percentage scores across categories, or the mean scores across UX variables. All these scores have been based on an unweighted sum of UX variables. Let us focus now on determining a UX index, which would be a unidimensional representation of ergonomic (EQ) and hedonic variables (HQ), or in other words, the UX level of OSDD systems (measured indirectly by EQ and HQ variables). This time, UX index would be the result of the weighted sum of UX variables, with weights determined by Factor and Principal Components analyses.

Table 28 shows the resulting components (Eigenvalues > 1) of the factor analysis for EQ and HQ variables. Component 1 explains 43.35% of the variance, whereas components 2 to 5 explain 8.72%, 6.34%, 5.69%, and 4.82%, respectively. It can be observed that most variables load towards component 1, thus suggesting they measure a single construct. Cronbach's alpha coefficient for EQ and HQ variables is 0.94 (high scale reliability). Both analyses suggest that EQ and HQ variables represent a single component. Hence, I will use the first Principal Component (PC1) score as the optimal unidimensional representation (linear combination or weighted sum) of my EQ and HQ variables in order to determine the UX index. Thus, UX index would be defined as the weighted sum of mean centered data for EQ and HQ variables. The loadings in Table 29 represent the weights, determined by the Principal Components Analysis.

The survey for users collected their perceptions of the appeal of OSDD systems. Table 30 shows the results of factor analyzing appeal variables. The factor analysis extracted a single component that explains 69.39% of the variance. The Cronbach's alpha for appeal variables is 0.90, which suggests that appeal variables measure a unidimensional component. Let us call Appeal index the representation of appeal variables. Thus, Appeal index would be defined as the weighted sum of mean centered data for appeal variables. The loadings in Table 31 represent the weights.

Table 28. Factor Analysis for Ergonomic and Hedonic Variables.
(Extraction Method: Principal Component Analysis)

Variables	Component				
	1	2	3	4	5
Creativeness	0.82				
Originality	0.80				
Impressiveness	0.80				
Modernity	0.78			-0.33	
Innovation	0.78				
Clarity	0.74	-0.30	-0.31		
Efficiency	0.73	0.36			
Controllability	0.72				
Motivation	0.70		0.40		
Simplicity	0.69		-0.35		
Security	0.69				
Accessibility	0.68				
Internal Consistency	0.66			0.43	
Exclusivity	0.65		-0.46		
Grouping	0.64	-0.35		0.35	
Interest	0.63		0.58		
Learnability	0.61	-0.56			
Expectation	0.59	0.53			
Excitement	0.57			-0.30	-0.36
External Consistency	0.53			0.49	-0.34
Supportiveness	0.57	0.65			
Value	0.42	0.46		0.36	0.32
Predictability	0.34	0.40			
Findability	0.38		0.32		0.67

Table 29. Loadings for Principal Component 1 on Ergonomic and Hedonic Variables.

PC1	0.13	0.19	0.10	0.10	0.29	0.11	0.24	0.18	0.21	0.26	0.13	0.18	0.16	0.19	0.25	0.29	0.26	0.28	0.05	0.23	0.15	0.20	0.17	0.26
Variable	Expectation	Security	Supportiveness	Predictability	Accessibility	Findability	Controllability	Learnability	Clarity	Simplicity	Internal Consistency	External Consistency	Efficiency	Grouping	Innovation	Modernity	Creativeness	Originality	Value	Excitement	Interest	Motivation	Exclusivity	Impressiveness
	EQ														HQ									

Table 30. Factor Analysis for Appeal Variables.
(Extraction Method: Principal Component Analysis)

Variables	Component
	1
Friendliness	0.91
Likeability	0.90
Enjoyment	0.88
Attractiveness	0.86
Goodness	0.82
Aesthetics	0.82
Desirability	0.54

Table 31. Loadings for Principal Component 1 on Appeal Variables.

PC1	0.28	0.34	0.51	0.35	0.35	0.53	0.15
Variable	Goodness	Likeability	Attractiveness	Friendliness	Enjoyment	Aesthetics	Desirability

The weights obtained for ergonomic and hedonic variables, as well as those obtained for appeal variables present only small differences. This suggests that an unweighted summation of variables for determining UX index and Appeal index would have also led to a high correlation between both indices. The correlations among unweighted scores for UX and Appeal, UX index, and Appeal index are:

- Correlation between UX index and Unweighted Appeal score: 0.89.
- Correlation between Unweighted UX score and Unweighted Appeal score: 0.89.
- Correlation between Unweighted UX score and Appeal index: 0.87.

Correlation between UX Index and Appeal

In order to measure the correlation between UX index and Appeal, I measured the correlation between UX index and Appeal index. The correlation analysis shows that both indices are strongly correlated (ρ=0.87 and p-value = 3.74189E-21). Hence, the UX level represented by the UX index and indirectly measured through ergonomic and hedonic qualities is strongly correlated to the Appeal or Appeal index directly measured by the survey for users.

UX Opportunity Areas

The analysis of survey results has shown that there are areas for improvement, which should be analyzed on a case by case basis. However, in order to narrow down the scope of opportunity areas (OA) to be analyzed in this thesis, it becomes crucial to define a criterion to spot the most relevant ones. Let us establish that the most crucial OA will be those for categories that scored less than the bottom quartile (Q1). The same would apply for variables that scored less than their respective UX dimension bottom quartile (Q1). Only ergonomic and hedonic qualities will be taken into account for spotting the OA since, as seen in the literature review, appeal is a trade-off between them. In order to improve the appeal of a system, its ergonomic and hedonic qualities have to be improved first. The OA will be analyzed by category of systems (see Table 5) and by UX variable (see Table 4) for the whole dataset.

Per Category

For users

Table 32. Opportunity Areas for Users by Category.

Category	Score	UX Dimension	Variables
OMA	75.00%	Ergonomic Quality	Efficiency (4.75), accessibility (4.63), simplicity (4.13), and external consistency (4.13)
	71.07%	Hedonic Quality	Impressiveness (4.63), creativeness (4.63), and excitement (4.00)
SOMB	70.92%	Ergonomic Quality	Controllability (3.50), external consistency (3.00), and grouping (2.50)
IOMA	67.35%	Ergonomic Quality	Security (4.00), Accessibility (3.60), Learnability (3.40), and Simplicity (3.00)
	63.71%	Hedonic Quality	Innovation (3.40), Excitement (3.40), and Modernity (3.00)
DERA	68.37%	Hedonic Quality	Modernity (4.00) and Excitement (3.43)

For developers

Table 33. Opportunity Areas for Developers by Category.

Category	Score	UX Dimension	Variables
SOMA	76.36%	Ergonomic Quality	Findability (4.83), external consistency (4.83), complexity (4.50), and accessibility (3.33)
FPB	70.41%	Ergonomic Quality	Accessibility (2.00)
	71.43%	Hedonic Quality	Modernity (3.00)
MPFAB	68.11%	Ergonomic Quality	Expectation (4.00) and Accessibility (3.00)
	81.79%	Hedonic Quality	Motivation (4.75) and innovation (4.25)
OMB	77.14%	Hedonic Quality	Modernity (3.50) and excitement (3.00)

Overall

For users

Table 34. Opportunity Areas for Users by UX Variable.

UX Dimension	Variable	Score
Ergonomic Quality	Controllability	4.30
	External consistency	5.34
	Simplicity	5.19
	Accessibility	4.92
Hedonic Quality	Creativeness	5.09
	Modernity	5.05
	Excitement	4.30

For developers

Table 35. Opportunity Areas for Developers by UX Variable.

UX Dimension	Variable	Score
Ergonomic Quality	Learnability	5.21
	Complexity	4.96
	External consistency	4.71
	Accessibility	3.43
Hedonic Quality	Creativeness	5.07
	Originality	5.07
	Modernity	4.18

List of UX Variables with Opportunity Areas

The discussion of OA will be based on the UX variables regardless of whether they relate to OA spotted by users or developers. Table 36 shows the aggregated list of variables that present OA according to users and developers. Variables in this list either scored below Q1 (3.95 and 3.81, for ergonomic and hedonic qualities, respectively) of mean scores, or their frequency was above Q3 (2.5 and 4, for ergonomic and hedonic qualities, respectively) of frequencies.

Table 36. Variables with Opportunity Areas.

UX Dimension	Variable	Mean score	Frequency
Ergonomic Quality	Accessibility	3.56	7
	Controllability	3.90	2
	Grouping	2.50	1
Hedonic Quality	Excitement	3.63	5
	Modernity	3.79	6

Other variables that were close to Q1 and Q3 were: expectation (4.00, 1) and security (4.00, 1) for ergonomic quality, and innovation (3.83, 2) for hedonic quality. Read numbers in parenthesis as (mean score, frequency). I consider important to discuss ergonomic variables: external consistency, learnability, predictability, and simplicity. I will also touch upon hedonic variables: innovation, creativeness, impressiveness, and originality.

Chapter 6. Discussion

This discussion will include recommendations to improve ergonomic and hedonic qualities based on the UX variables. The variables here discussed were identified as opportunity areas (OA) for the OSDD systems, according to the surveys' results analysis. This chapter is not intended to be a comprehensive and strict set of rules for improving the user experience, but rather a set of recommendations based on standards and my own experience in the user interface design terrain. When possible, I will refer the reader to other resources that contain more information about the topic being discussed.

Ergonomic Quality

Let us remember that the ergonomic quality is related to do-goals. Users do things every day to accomplish their work. The ergonomic quality in turn is instrumental to the hedonic quality or be-goals. Users within the Bank do things for many underlying reasons (e.g., be helpful to others, be an outstanding employee, be responsible, be helpful to the country). In order for users to accomplish their be-goals, systems must first provide a platform for accomplishing do-goals (e.g., do data analysis, write a report, conduct an auction). For this reason, systems should excel at ergonomic quality. The analysis of survey results showed that the following ergonomic attributes had higher priority as opportunity areas.

Accessibility

Of the world's population, 15% live with a form of disability [20]. In Mexico, 8% of men and 2% of women have color blindness [21]. It is logical to think that the Bank's users are no exception. With around 80 users of systems, at least 5 could be color-blind. That is why developers should account for accessibility when building systems. Two types of disabilities that normally go unnoticed are color- blindness and hearing impairment. Let us address color blindness first. People with this form of disability have a hard time trying to make sense of information that is categorized by color, especially with green and red color spectrums. But some people have problems with yellow and blue color spectrums, too. Figures 25 to 29 illustrate how people with different color blindness see, compared to normal vision [22][23][24].

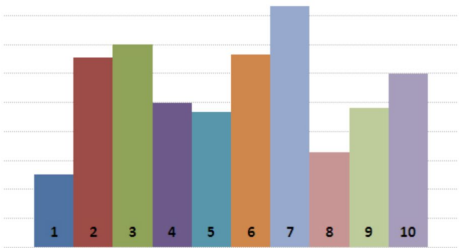


Figure 25. Normal vision.

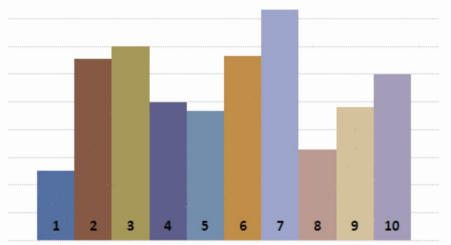


Figure 26. Deuteranomaly (green-weak).

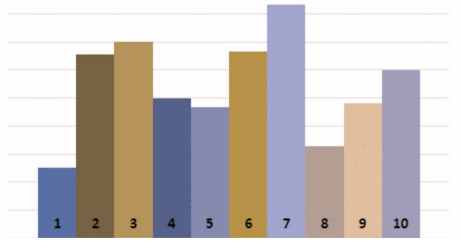


Figure 27. Deuteranopia (green-blind).

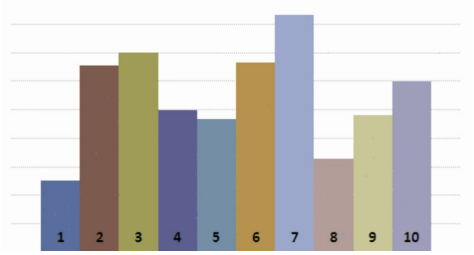


Figure 28. Protanomaly (red-weak).

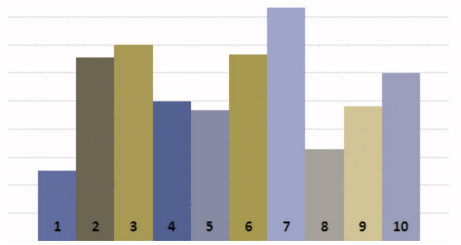


Figure 29. Protanopia (red-blind).

In order to take color-blind users into account, developers should design with color-blindness in mind. Here are some recommendations developers can follow in order to make their systems more color blind inclusive [25]:

- Before designing, choose a color scheme that can be identified by everyone. See [26] for some color-blind-friendly palettes.
- Do not focus only on colors for categorizing data. It is possible to play with textures, positions, font sizes, line types, symbols, etc., in order to make sure color-blind people can understand the message.
- Do not use the following color combinations to contrast information: red and green, green and brown, green and blue, blue and gray, blue and purple, green and gray, and green and black.
- Use highly contrasting dark and light colors. Most color-blind people can perceive high contrast.

Regarding hearing impairment, developers should provide visual aids for attracting a user's attention. For example, blinking titles for alerts, or flashing windows every time a sound is generated by the interface. In general, it is a good practice to incorporate both visual and sound alerts.

There are many other ways to improve a system's accessibility. Please refer to [27] for more information on this topic.

Controllability

Interfaces should provide users with graphical components that are easy to control. Sometimes users struggle entering dates, decimal figures, credit card numbers, email addresses, etc. Developers should not only fulfill do-goals, but they should also account for friendliness and efficiency. Some recommendations for building more controllable interfaces are:

- Group information of the same nature (see grouping).
- Provide *drop-down* lists for pre-existing catalogues (countries, neighborhoods, gender, etc.)
- Provide users with autocomplete features in text fields whenever possible.

- Use specialized graphical components for specific fields, such as dates, decimal numbers, and emails (date picker, numeric field, email keyboard, etc.).
- Add a *reset* option in case users want to go back to the original values.
- Add a *select/deselect all* option wherever it makes sense.
- Add drag-and-drop functionality wherever it makes sense.
- Do not make users refill fields that ask for information that was entered in previous steps.

Grouping

Information of the same kind should be grouped, thus easing the selection or interpretation process for users. Imagine if the numbers on a calculator were spread out over its board. It would be hard to enter figures and the calculator would be error-prone. Objects can be grouped by proximity, similarity, continuity, closure, area, and symmetry [30].

Objects that are closer to each other tend to be seen as a single group (see Figure 30). White space is an underestimated tool that can help group objects by proximity. Simply add more space between items that are not related and reduce white space between items of the same sort. The use of white space can help to keep an interface's neatness by removing the need of solid lines for grouping things.

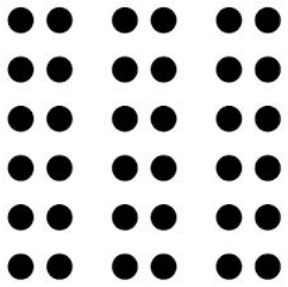


Figure 30. Grouping by proximity.

It does not matter if many objects occupy the same space — the ones that are similar will be clustered by users (see Figure 31). Similarity could be expressed by color (red, green, blue, etc.), shape (square, triangle, circle, etc.), texture (dotted line, soft surface,

hard surface, etc.), size (small, medium, large, etc.), or spatial orientation (inclination, rotation, vertical or horizontal alignment, etc.).

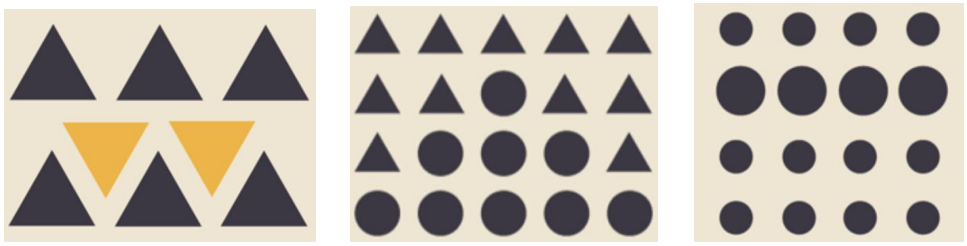


Figure 31. Grouping by similarity.
(left) Color; (middle) Shape; (right) Size.

Objects that are laid out in a contour, tend to be grouped by continuity. All the stones in a cobblestone path that connects the main entrance of a house to the street, through the front yard, are not seen as single stones, but as a cobblestone path. This is because all the stones follow the same contour. Another example of grouping by continuity is a group of points in a chart that represent a series of data. The human eye groups all points into a single curve (contour) that represents a specific phenomenon being measured. Finally, a cross will be interpreted as two intersecting lines rather than as four lines meeting at a point. Human eyes like continuity and group objects accordingly. Figure 32 illustrates grouping by continuity.

Figure 32. Grouping by continuity.
(left) The eye sees two intersecting curves rather than a series of points.
(right) The eye sees a cobblestone path rather than a series of stones.

Grouping by closure refers to how the eyes tend to perceive complete and closed figures, even when lines are missing. Figure 33 illustrates grouping by closure. It can be

noticed how the eye tends to see a square rather than just four semi-full circles, even if the square contour is missing some lines.

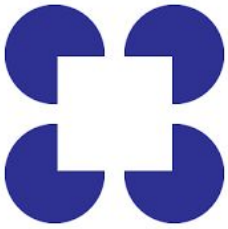


Figure 33. Grouping by closure.

When two or more elements overlap, we tend to see them as one on top of the other, rather than all being part of a single object. This is called grouping by area. In Figure 34, the eye tends to see a circle with a triangle on top, rather than a circle with a hole with the shape of a triangle.

Figure 34. Grouping by area.

The eye prefers greater symmetry when it comes to interpreting. In Figure 35, the eye tends to see two overlapping squares, rather than three polygons.

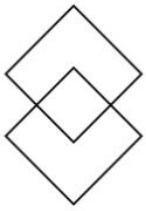


Figure 35. Grouping by symmetry.

Always group objects or information of the same kind taking into account how the eye will perceive them. This will prevent users from receiving the wrong message.

Expectation

Users have needs, which sometimes translate to a new system development request. At the beginning of a new system development project, programmers gather user requirements in order to know what the user expectations are regarding the new system. Functional requirements refer to what the system should do, whereas non-functional requirements are more related to how and why the system should do what it is supposed to do.

Developers are used to cope with functional requirements since they are easier to validate than non-functional ones. This is due in part to non-functional requirements being related to the be-goals, which developers tend to see as subjective. This subjectivity inclines programmers to ignore be-goals, thus causing systems not to meet user's expectations entirely.

Users could ask for a functionality to display the percentage of market makers involvement in the current month. Its counterpart, the non-functional requirement (be-goal), could be displaying the information in a way that makes it easy to make very quick decisions. The first part is easy to interpret, whereas "in a way that makes it easy to make very quick decisions" is not. Programmers could choose different methods for displaying the requested information (e.g., a table, a chart, a list). Which option will allow users to make quicker decisions? Well, it depends on the user. Some might be very good at reading lists and get insight about the data very quickly, while others might be more visual, thus preferring charts. There is no right or wrong answer when it comes to fulfilling be-goals. The key is asking the actual users of the system what they would prefer.

Development engineers who have seen users operate other systems, could have a better idea of how these users would like a functionality to be implemented. However, the final decision should come with the user's consent. This will prevent users from rejecting some functionalities or the entire system in the future.

Here are some recommendations for developers when it comes to accounting for be-goals:

- Try to know the "why" behind a requirement before getting to the "how". Get

informed about why users asked for something. Once you know the “why”, it will be easier to get to the right “how”.

- Do some brainstorming of possible solutions before starting to code and do not jump to the final implementation without having users’ approval of your solution.
- Make prototypes and show them to users. A picture is worth a thousand words. By using prototypes, it will be easy for developers to convey their ideas to users. Also, it will be easy for users to accept or reject a specific implementation.

  Do not spend too much time making the perfect prototype. Follow the “fail fast” philosophy by doing low definition prototyping such as paper prototyping, which you can show to users in a matter of hours or less. Thus, the development team will be able to get user feedback faster and move on. For a more sophisticated prototype, try combining paper prototyping with apps that allow transforming the paper design into an interactive prototype (e.g., POP). Another option is to use apps that allow building digital low fidelity prototypes (e.g., Balsamiq).

  Once you are done with the low-fidelity prototype, try building a high-fidelity one. Use apps that provide you with graphical components that are the same or very similar to those used by the GUI tool or programming language being used by the development team (e.g., Axure RP, Gravity Designer). Some tools even generate the final GUI code automatically. High-fidelity prototypes will allow users to spot misalignments hard to see in low-fidelity prototypes.

- Never overlook a user’s opinion whenever you can ask for it — no matter how experienced you think you are. The user experience is characterized by its temporality. Things that work or are seen a certain way today, can work or be seen completely differently tomorrow. Experience is good, but it should not stop us from going to the source.
- Never ignore a non-functional requirement (be-goal) again if you want to comprehensively fulfill users’ expectations.

Security

It is normal that users make errors when using interfaces. These errors are potentiated when interfaces are poorly designed. Errors can be classified into slips and lapses, and mistakes depending on the context in which they occur [32]. Slips and lapses happen

when users make errors when executing well-known procedures (pressing an onscreen button, opening a dropdown menu, etc.). If a user clicked before being over the button, the error is called a slip. Another example of a high-level slip would be attaching the wrong file to an email or not attaching it at all.

The difference between slips and lapses resides in the origin of the error. A slip would happen when a user substitutes an action for another in a procedure (pick the wrong file or clicking accept before selecting the file for email attachments), whereas lapses occur due to a failure of memory (forgetting the overall goal, forgetting the step you are in the execution of a procedure, etc.). Mistakes, on the other hand, are related to rule-based (application of learned if-then rules) or knowledge-based (problem-solving, logic, experimentation, etc.) behavior, being less common than slips or lapses since users spend most of their time executing well-known procedures than engaging into tasks that require further concentration and experimentation [32]. Figure 36 shows the types of errors.

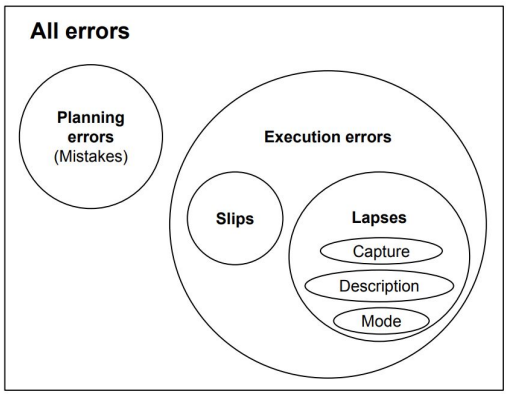


Figure 36. Types of errors.

Developers should design interfaces taking into account that users might commit errors (for different reasons) when working with them. The design should prevent as many errors as possible. Here is a list of recommendations to make the interfaces less error-prone:

- Avoid very common action sequences that have identical prefixes in order to prevent capture slips, which are errors committed when capturing commands that are used very often and that have similar patterns (executing `:wq` instead of just `:w` in vi, UNIX).

- Different things should look and act differently in order to prevent description slips, which are errors committed when users select one object over another because they look very similar or project the same affordances (e.g., an email interface with a long row of buttons that make it easy to click on the delete button, when the answer option is the one the user wanted to select).
- Keep dangerous functions separate from frequently used ones in order to prevent a huge negative impact in case users make description slips.
- Provide, whenever possible, a way to recover from description slips (e.g., undo, redo). Only when recovering cannot be provided for a very dangerous function, developers should use confirmation dialogs in order to prevent unintentional executions. Overuse of confirmation dialogs should be avoided as users could become used to them and passively ignore them, thus increasing the risk of big fails.
- Eliminate modes or make users know clearly that they are working on a specific mode, in order to prevent mode slips, that is, errors that occur when users try to execute the same command in a different context or mode (Caps Lock mode, formula typing mode on a spreadsheet, etc.). It is very usual to type an uppercase letter instead of a lowercase one when the Caps Lock mode is activated since awareness that users are working on that mode is not on the users' locus of attention. Interfaces should always let users know what mode they are working on or try to avoid the use of the same commands in different modes in order to prevent users from getting confused.

External Consistency

As we start using a new system, we tend to look for known patterns such as the file menu to save our progress or the brush icon to copy and paste the format of a text or a cell in a spreadsheet, a very well-known feature for Microsoft Office users. Another example of a well-known pattern is the tooltip message that is displayed when the mouse hovers over a button, thus indicating with a short message its functionality.

People have knowledge they have acquired during their lives that helps them lower the learning curve when utilizing new tools or devices. Graphical Interfaces should be designed taking external consistency into account. Users will adapt more quickly when new systems incorporate a wide array of standard functionalities. Therefore, it is

important for developers to adopt design patterns external to their work that belong to the interface design discipline [28].

When designing custom-made systems, sometimes it is necessary to create new graphical components from scratch to address a specific set of needs. The Bank has many specific needs and it is totally acceptable to custom-build. In fact, not all functionalities have to rely on standards, since there are not enough standards to cover all specific needs.

Humans are creative beings who like to innovate. During this innovation process, they design new ingenious solutions for common or new problems they face every day. Since standards require consensus, they move at a slower pace, thus being unable to keep up with human innovation speed. This does not mean engineers should overlook standards. A trade-off between standards and custom-made solutions should be made in order to satisfy users' expectations while helping them lower their learning curves when interacting with a new system.

It is worth clarifying that the mere fact that an interface incorporates standards does not make the entire interface usable by default [31]. In the end, user testing must be done in order to determine how usable an interface is. It remains important for systems to be externally consistent when it comes to very well-known standard solutions that can help to solve a requirement in a more efficient and less error-prone fashion.

Please refer to [29] for more information on standard patterns for interface design. Even though these patterns focus on web interfaces, the graphical components they touch upon (e.g., accordion, button, carousel, checkbox, combo box) are the same that are used in standalone interface design, such as desktop or mobile.

Learnability

It is important to lower the learning curve of users when it comes to interacting with new systems. When users interact with a system's interface for the first time, they tend to assume the interface has certain affordances (e.g., I can click on it, I can close it, I can save my progress, I can export information). These pre-constructed affordances come

from users' experiences with other interfaces.

Systems that do not provide standard affordances are harder to learn. Even if systems do provide them, if they do not incorporate the right signifiers[14] (e.g., floppy disk button to save progress, pdf icon for exporting information, an x button to close), users will still have a hard time adapting. Systems that account for external consistency and grouping, are easier to learn than those that do not.

For in-house systems, it is important to account for internal consistency. That way, the affordances and signifiers users have learned for a certain system, would also apply to other systems within the Bank.

A user's first impression of an interface will shape the way he or she perceives the system as a whole. Systems that are easy to learn, motivate users to keep using them and exploring them. In contrast, when users have a hard time understanding how to start using systems, they tend to stop trying and move on. I remember once seeing two guys in the gym trying to use a bicycle that not only helped you exercise but also had a pair of glasses that allow you to live virtual reality experience while pedaling. To start, just put the glasses on, the machine instructions said. The guys put on the glasses and the virtual reality screen never started up. After restarting the machines, the system still did not work. I just saw the guys giving up and leaving. The same can happen with the operating systems within the Bank. This is why developers should design intuitive interfaces that would make it easy for users to start using new systems.

Predictability

When users click on a button and the system ends up doing something completely unexpected, they tend to distrust such a system and the development team. It is important that systems behave in a predictable manner — that is, users should see the same results when executing the same task over the same dataset. The development team, as well as the testing team, should test multiple and varied scenarios in order to make sure systems will behave consistently all the time. In order to spot possible misalignments, it is advisable to test on the extremes (holidays, first and last days of the

[14] Signifiers indicate how a system affords a certain functionality. For example, the affordance could be "this jar can be opened" and its related signifier could be a visible lid the user could turn to the left in order to open the jar.

month, first and last day of the year, division by zero, minimum possible value, maximum possible value, etc.) and not testing only the most frequent scenarios.

A common design mistake is to allow users to enter information through a succession of steps and let them know the information they entered is incorrect, only at the end. Users expect that if a system allowed them to enter a certain figure or text and did not warn them of an error right away, they have entered it correctly. Users cannot predict a system will stop them just after they have invested some time trying to execute a task.

The development team should design and test in order to account for predictability. If they do it, users will build trust in them as well as in their systems.

When users inform developers about unpredictable behaviors, developers should take the report very seriously and not just dismiss it by treating it as a consequence of a user's inexperience with the interface.

Simplicity

It is hard for users to propose new functionalities or improvements for systems when they do not understand how systems work. Some systems have been around for decades. New users that did not participate in the design of a system, might see it as a black box, which helps them do their work, somehow users do not totally understand. This is why developers should care about providing any kind of help in order to make processes look transparent. User manuals as well as on the spot help (custom tooltips that include text and images to explain how a process is done) are good transparency tools.

Examples of information that users should have at hand in order for them to be able to understand how the system works are: the algorithms that are being used for executing a set of processes, the rules that are being followed when rounding figures up or down, the convention used regarding the number of decimal spaces used to present different kinds of figures (rates, prices, currency), etc. If users understand how the system works, they will be able to propose better ways systems could process information or present results on screen, thus improving the overall quality of systems as products.

Hedonic Quality

This quality relates to be-goals, which represent the underlying reasons behind users' requirements. When developers understand the why (to build something) behind the what (to build), it is easier for them to determine the how (to build something). The interface design should take into account this UX dimension in order to improve the appeal of systems, as perceived by users. The following UX attributes represent the opportunity areas for the hedonic quality according to the analysis of survey results.

Excitement

The Oxford dictionary defined excitement as "A feeling of great enthusiasm and eagerness". It seems like the computer systems used in the Bank are not supposed to evoke excitement on users. Developers cannot expect to see users excited all the time by using this or that functionality. However, the user experience that computer systems generate can be accompanied by excitement. Excitement can be a manager feeling amazed by how easy it is to achieve greater things by a single click, an analyst that could easily get insights from information thanks to the way interfaces convey it on screen, an employee who can go home earlier thanks to how fast the software executed a process that used to take weeks, a user that manages to understand a system so well that feels eager to improve it, etc.

Let us not take the lack of excitement for granted in computer systems developed for a bank and let us be proud when seeing users feeling excited when interacting with our systems. Sometimes, in order to see it, you have to believe it. In order to see excitement, we have to believe our systems can evoke it.

Modernity

Modernity feels good, it feels fresh, and it feels subjective. It is good for systems to look modern and stay relevant to users. Modernity is not only about impressive aesthetics, but also about new methods for doing things such as new algorithms for improving processing times or new interface layouts that are more usable, thus making a system look up-to-date and friendly.

The same way people choose what kind of clothes fit them, developers should also analyze whether a new mode or trend will add or rest value to their systems. It does not seem to be a good practice to incorporate new trends just because they will make a system feel more modern. The cost-benefit relation should always be taken into account. Let us remember that systems that do not deliver value are abandoned by users, and the same happens with commonly known modern functionalities. Sometimes less is more. Keeping interfaces simple is not easy, but it is something developers should strive for in order to make users' lives easier and do not overwhelm them with features that do not add value to their work. Modernity should not be regarded as something good or bad, but as a tool that can bring value to systems when used properly.

Innovation

The word innovation evokes change. In computer systems, innovation topics developers might talk about include innovative algorithms, more powerful storage systems, retina screens, 3D printing, etc. We just got used to Gigabytes when Terabytes and even Petabytes are making their way through. Innovation can sound scary and exciting at the same time. Developers want to innovate, as they do not want to be left behind. However, keeping the pace of innovation is exhausting, especially nowadays when innovation seems to be around the corner.

Innovation does not have to be completely disruptive to be considered as such. Sometimes, small changes can have a huge positive impact on users' work. Innovation and creativeness are not disjoint concepts. Creativity can lead to innovation, and innovation can foster creativity as new knowledge is acquired.

Programmers should not feel afraid of innovation and should even foster it. They do not have to invent a new and sophisticated algorithm in order to innovate. Simple changes such as displaying the information in a more intuitive way (even is that kind of information had always been displayed in a certain way), adding a tooltip for informing the purpose of a certain graphical component, or just slight rearrangements in the layout of interfaces, can have a huge impact.

Small innovations can lead to great innovations eventually. Let us start innovating on a small scale in order to achieve bigger goals in the future.

Creativeness

Developers should always try to find clever solutions that do more with less. In computational terms, this could mean less memory usage, less processing time, fewer steps for users in order to execute a process, etc. Human beings are creative by nature, they are always trying to improve the current state-of-the-art by optimizing existing algorithms or simplifying tasks to users. Creativity seems to be undervalued in the engineering world, when in fact creative minds should be fostered for the sake of progress. Developers should start fostering creative minds. There will always be resistance to change, resistance to lose the status-quo. However, the world keeps moving and sooner or later new and creative ideas will end up substituting the current ones. Why not, instead of resisting creativity, we foster it and be part of the change?

Creative minds face resistance in a world full of egos. It is like a competition that focuses on establishing whose ideas should prevail. The most creative ideas can be found where less expected. That a person does not hold a specific degree does not make his or her ideas less creative or less valuable than those that come from experts. Rejecting ideas without analyzing them just because they seem unfeasible at first sight, could discourage creative people. Instead of saying no, why not asking for a more detailed proposal about how the idea could be implemented. The user experience is temporal and evolves all the time. What today seems best, might seem inferior tomorrow. Let us evolve along with the user experience and reach higher horizons. Let us welcome creativity in our daily lives.

Impressiveness

The Oxford dictionary defines impressive as “Evoking admiration through size, quality, or skill; grand, imposing, or awesome”. Computer systems are pieces of software designed to improve people’s life (most of the times) at work, school, or home, and even at a personal level. Experienced engineers tend to overlook how hard it is to build new software. What seemed to be impressive years ago might seem nondescript now. This can happen for users as they get used to the highest standards.

Computer systems become commodities as time goes by. In order for systems to keep being impressive for users, they have to evolve at the same pace technological trends do. This might seem overwhelming from a developer's perspective since new technologies seem to appear every day. Ultimately, a trade-off between keeping a healthy workload for users and developers, and the pace at which new technologies are incorporated to current systems or new systems has to be done.

A system does not have to incorporate all new trends in order to keep its impressiveness. What technologies should be incorporated or skipped is a decision that has to be made by prioritizing users' needs. The right balance between cost and benefit of incorporating a new solution has to be pursued.

The top management should always bear in mind that developers need to get the best training in order for them to be able to incorporate new technologies to existing software properly. It does not matter how available a technology is and how good it might be when it comes to fulfilling users' needs if the development team does not have the expertise to handle it.

Programming is an art, and like any art, it has its own rhythm. Great artists achieve great works of art when they feel inspired. Inspiration comes in part by observation. When we observe we acquire new knowledge and expand our possibilities. For this reason, developers should be given time and encouraged to do research, take professional training, and observe users' work, all as part of their own work. This would provide them in turn with the tools that innovation requires.

Let us not forget that developers and users are human beings, as people in charge of making impactful decisions are, too. Let us act all in the most humane way when it comes to building computer systems, too.

Originality

Originality is closely related to uniqueness. Developers feel proud when they build a new feature that is one of a kind. This feature will have the developer's DNA on it. Users might even refer to this feature by the developer's name or the developers' team name. Human beings spark uniqueness and value things that are original. However, the need to push people to be original might become a hassle. People can even feel demotivated

when they are not capable of delivering original things. The top management should take this into account and do not undervalue developers work just because it does not seem to be original.

As for innovation, originality does mean doing everything based only in our own creativity or ideas. In the end, the knowledge people have has come from other sources. New knowledge can be generated by embracing other people's opinions, first. Developers should not be afraid to develop new solutions just because these are not entirely based on their own knowledge. Even geniuses rely on other people's work to add originality to existing products or systems. The most renowned scientists rely on other scientists' work as well as on their own original work in order to innovate.

Originality does not have to be a synonym of uniqueness as a whole. People can contribute their originality at any scale. Maybe a developer decided that a new and original graphical component that is a small part of a system can make data analysis easier for users. Developers do not need to build a wholly unique system in order to show originality, as they can show originality at the feature level. Let us accept that as humans, we are unique, and this uniqueness leads us to spark originality through creativity.

Chapter 7. Summary and Next Steps

During the writing process of this thesis, my definition of user experience has radically changed, from equating it to usability to seeing it as something holistic that involves not only the hows but also the underlying whys. I hope that it will happen the same to all readers.

I learned and could support the hypothesis that states that the user experience is the result of a trade-off between ergonomic and hedonic attributes. This trade-off will in turn determine the appeal of a systems from a user's perspective. The UX is not something that can be measured directly. It is a construct that results from the evaluation of many attributes: ergonomic (expectation, accessibility, learnability, etc.), hedonic (modernity, originality, excitement, etc.), or whatever kind that might be suggested in the future.

My goal was to measure the UX level of operating systems within the Central Bank of Mexico. In order to do this, I developed a user experience framework (UXF) that allowed me to determine the UX attributes I should use to ask for users' opinion regarding the user experience level they have had when interacting with the Bank's operating computer systems. To collect users' perceptions, I decided to use a questionnaire or survey based on the semantic differential technique (polar verbs) with a seven-point scale. In order to facilitate the scoring process to users of the different UX attributes, I decided to describe how I would interpret each number in the scale. This idea was also suggested by people who tested the survey, as something that could improve the experience of users when responding to the survey.

I applied two surveys in the Bank, more specifically within the Directorate General of Central Bank Operations and Payment Systems. One survey was for users of systems and the other for developers of systems. The surveys allowed me to contrast the level of the experienced users had when using systems, to what developers think the users' perspective is. I was happy to see that both perspectives are aligned, with only a few significant differences.

The survey results were analyzed by category of operating computer systems and as a whole. The analysis by category showed that for users, systems within categories FPB, MPFAA, MPFAB, OMB, and FPA have the highest appeals (80% or more). For these categories, the ergonomic attributes: efficiency, expectation, findability, grouping, internal consistency, simplicity, and supportiveness, were scored highest. On the hedonic side, the attributes: exclusivity and value, were scored highest. Users found that aesthetics, desirability, friendliness, goodness, and likeability were the most appealing characteristics for these categories.

In contrast, categories DERB, DERA, OMA, and IOMA obtained the lowest scores (less than 80%). The ergonomic variables: clarity, external consistency, learnability, predictability, and simplicity, scored lowest. In the hedonic terrain, attributes: creativeness, excitement, impressiveness, innovation, and modernity, ranked lowest. Users think that aesthetics attractiveness and enjoyment are the less appealing characteristics for these categories.

Factor and Principal Components analyses were used in order to determine whether UX attributes measure a unidimensional measurement, called the UX index or UX level (measured through ergonomic and hedonic attributes). The same was done for Appeal or Appeal index. It was determined that attributes indeed loaded to a single component both for UX index and Appeal index. Cronbach's alpha for ergonomic and hedonic, as well as for appeal showed high scale reliability, which again suggests unidimensionality for ergonomic and hedonic qualities, and appeal. Finally, it was determined that the UX index was highly positively correlated to the Appeal index, thus supporting the hypothesis that the UX level is a trade-off between the ergonomic and hedonic qualities.

Some limitations regarding of the analysis are:

- A small dataset size.
- Due to the small data size, only simple statistical methods such as quartiles were used in the analysis of survey results.
- Other data collection such as user and developers interviews, could have been done too.

Next steps could include, but are not limited to:

- Plan of action for implementing recommendations.
- Further factor analysis of ergonomic and hedonic attributes in order to determine whether they fit the categories proposed by the UXF (dependability, perspicuity, efficiency, novelty, and stimulation) or a different classification should be proposed.
- More research in other central banks in order to increase dataset size and determine UX commonalities of computer systems that support daily operation in such incumbent institutions.
- Further research to determine whether there are more UX dimensions than those proposed in the UXF (ergonomic quality, hedonic quality, and appeal), and those proposed by the authors hereby analyzed in the literature review.

References

[1] Hassenzahl, Marc, and Virpi Roto. *Being and Doing: A Perspective on User Experience and Its Measurement*. Vol. 72, 2007.

[2] "ISO 9241-210:2010(En), Ergonomics of Human-System Interaction — Part 210: Human-Centred Design for Interactive Systems." Accessed March 26, 2019. https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en.

[3] Law, Lai-Chong, Virpi Roto, Marc Hassenzahl, Arnold Vermeeren, and Joke Kort. *Understanding, Scoping and Defining User Experience: A Survey Approach*, 2009. https://doi.org/10.1145/1518701.1518813.

[4] Kashfi, Pariya, Agneta Nilsson, and Robert Feldt. "Integrating User EXperience Practices into Software Development Processes: Implications of the UX Characteristics." *PeerJ Computer Science* 3 (October 16, 2017): e130. https://doi.org/10.7717/peerj-cs.130.

[5] Hassenzahl, Marc. "The Thing and I: Understanding the Relationship Between User and Product." In *Funology: From Usability to Enjoyment*, edited by Mark A. Blythe, Kees Overbeeke, Andrew F. Monk, and Peter C. Wright, 31–42. Human-Computer Interaction Series. Dordrecht: Springer Netherlands, 2005. https://doi.org/10.1007/1-4020-2967-5_4.

[6] Wright, Peter, and John McCarthy. "Experience-Centered Design: Designers, Users, and Communities in Dialogue." *Synthesis Lectures on Human-Centered Informatics* 3, no. 1 (January 1, 2010): 1–123. https://doi.org/10.2200/S00229ED1V01Y201003HCI009.

[7] Jordan PW. 2002. Designing pleasurable products: an introduction to the new human factors. Boca Raton: CRC Press.

[8] NN group. *Don Norman: The Term "UX."* Accessed April 3, 2019. https://www.youtube.com/watch?v=9BdtGjoIN4E.

[9] "The 7 Factors That Influence User Experience." The Interaction Design Foundation. Accessed April 3, 2019. https://www.interaction-design.org/literature/article/the-7-factors-that-influence-user-experience.

[10] “What Are Affordances?” The Interaction Design Foundation. Accessed April 3, 2019. https://www.interaction-design.org/literature/topics/affordances.

[11] “2017 Disability Statistics Annual Report,” n.d., 48. Accessed April 3, 2019 https://disabilitycompendium.org/sites/default/files/user-uploads/2017_AnnualReport_2017_FINAL.pdf

[12] “Población Con Alguna Discapacidad En México Según El INEGI.” Accessed April 3, 2019. http://cuentame.inegi.org.mx/poblacion/discapacidad.aspx?tema=P.

[13] “Color Blindness Diagnosis.” *Colour Blind Awareness* (blog). Accessed April 3, 2019. http://www.colourblindawareness.org/colour-blindness/diagnosis/.

[14] Hassenzahl, Mare, Axel Platz, Michael Burmester, and Katrin Lehner. “Hedonic and Ergonomic Quality Aspects Determine a Software’s Appeal.” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 201–208. CHI ’00. New York, NY, USA: ACM, 2000. https://doi.org/10.1145/332040.332432.

[15] Hassenzahl, Marc. *The Effect of Perceived Hedonic Quality on Product Appealingness*. Vol. 13, 2001. https://doi.org/10.1207/S15327590IJHC1304_07.

[16] Mahlke, Sascha, and Manfred Thuering. *Studying Antecedent of Emotional Experiences in Interactive Contexts*, 2007. https://doi.org/10.1145/1240624.1240762.

[17] Law, Lai-Chong, Virpi Roto, Marc Hassenzahl, Arnold Vermeeren, and Joke Kort. *Understanding, Scoping and Defining User Experience: A Survey Approach*, 2009. https://doi.org/10.1145/1518701.1518813.

[18] Laugwitz, Bettina, Theo Held, and Martin Schrepp. “Construction and Evaluation of a User Experience Questionnaire.” In *HCI and Usability for Education and Work*, edited by Andreas Holzinger, 5298:63–76. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. https://doi.org/10.1007/978-3-540-89350-9_6.

[19] “What Is User Experience (UX) Design?” The Interaction Design Foundation. Accessed May 6, 2019. https://www.interaction-design.org/literature/topics/ux-design.

[20] “WHO | World Report on Disability.” WHO. Accessed May 18, 2019. http://www.who.int/disabilities/world_report/2011/report/en/.

[21] “Daltonismo, Una Enfermedad Masculina | Vanguardia.” Accessed May 18, 2019. https://vanguardia.com.mx/daltonismounaenfermedadmasculina-1345289.html.

[22] “Data Visualization That Is Colorblind-Friendly - Excel 2007?” Analytics Demystified, June 18, 2009. https://analyticsdemystified.com/excel-tips/data-visualization-that-is-color-blind-friendly-excel-2007/.

[23] “Deuteranopia – Red-Green Color Blindness – Colblindor.” Accessed May 18, 2019. https://www.color-blindness.com/deuteranopia-red-green-color-blindness/.

[24] “Protanopia – Red-Green Color Blindness – Colblindor.” Accessed May 18, 2019. https://www.color-blindness.com/protanopia-red-green-color-blindness/.

[25] “How to Optimize Charts For Color Blind Readers Using Color Blind Friendly Palettes.” *Venngage* (blog), June 25, 2018. https://venngage.com/blog/color-blind-friendly-palette/.

[26] “Data Visualization, Design and Information Munging // Martin Krzywinski / Genome Sciences Center.” Accessed May 18, 2019. http://mkweb.bcgsc.ca/colorblind/.

[27] “Accessibility - W3C.” Accessed May 18, 2019. https://www.w3.org/standards/webdesign/accessibility.

[28] Heydon. “Inclusive Design Principles: Be Consistent.” *TPG* (blog), August 11, 2017. https://developer.paciellogroup.com/blog/2017/08/inclusive-design-principle-be-consistent/.

[29] “WAI-ARIA Authoring Practices 1.1.” Accessed May 18, 2019. https://www.w3.org/TR/wai-aria-practices-1.1/#aria_ex.

[30] “Reading 14: Layout.” Accessed May 18, 2019. http://web.mit.edu/6.813/www/sp18/classes/14-layout/.

[31] “HCI Standards | ACM Interactions.” Accessed May 19, 2019. https://interactions.acm.org/archive/view/march-april-1999/hci-standards1.

[32] “Reading 4: Safety.” Accessed May 19, 2019. http://web.mit.edu/6.813/www/sp18/classes/04-safety/.

Appendix A. Survey for Users

English Version

Survey for Users Structure

- **Landing page**
- **Demographics** (3 questions)
- **Computer Systems Category Selection** (1 question)
- **Ergonomic Quality** (14 questions divided in two sections)
- **Hedonic Quality** (10 questions)
- **Appeal** (7 questions)
- **Open question** (1 question)

36 questions in total

If every question is answered in 15 secs (or 4 questions per minute), this survey should not take more than 10 min approx.

Survey for Users

The goal of this survey is to measure the current User Experience (UX) level of computer systems that have been developed by the *Operating Systems Development Division (OSD-D)* within the Directorate of Operations Support (DOS). This survey has been designed to be responded by all analysts, chiefs of division, deputy managers, managers, and directors that make use of such systems at work.
UX design is a new way of building systems that account for both *ergonomic* (e.g., effectiveness, learnability, efficiency) and *hedonic* (e.g., innovativeness, originality, aesthetics) aspects of software. Both aspects combine in users' minds and allow them to come up with a general appeal of systems. Any computer system delivers a user experience either positive or negative, regardless of whether UX design was taken into account or not.

This survey is part of Jonathan Laguna's thesis titled *Implementing a User Experience Policy for Operating Computer Systems Development in the Central Bank of Mexico*. Jonathan is currently enrolled in the System Design in Management Program (SDM) at MIT. He worked for the Central Bank of Mexico in the CBOS division for almost ten years as a systems development analyst before coming to MIT, and he is planning to continue working for the Bank after graduation.

This survey is intended to be anonymous. Only Jonathan and his thesis supervisor Dr. Eric Rebentisch will have access to the information here collected. Your responses will help spot opportunity areas for the improvement of operating systems development.

Survey for Users

Demographics

1. Please select the division you work for currently **SINGLE CHOICE**
   a. Directorate of Domestic Operations
   b. Directorate of International Operations
   c. Directorate of Operations Support

2. Select the number of years you've been working for the Bank **SINGLE CHOICE**
   a. 0 to 10
   b. More than 10 and up to 20
   c. More than 20 and up to 30
   d. More than 30

Demographics

3. Please select your background (s) **MULTIPLE CHOICE**
   a. Economy
   b. Actuary
   c. Accounting
   d. Finance
   e. Informatics
   f. Mathematics/Physics
   g. Computer Systems or related
   h. Other (don't have to specify)

Dependability

Think of the systems you selected previously for answering the following questions.

5. Do you consider they **meet your expectations**? (i.e., they do what they were expected to do and they do it the way you were thinking they would do it)

Meet expectations 1 2 3 4 5 6 7 Doesn't meet expectations

Dependability (continues)

Think of the systems you selected previously for answering the following questions.

6. Do you consider they are **secure**? (This is not about security against hackers but rather on their graphical user interfaces use. For example, an insecure system would be one where it's very easy to click on the wrong button causing information deletion or the execution of an operation that didn't have to be processed at all)

Not secure 1 2 3 4 5 6 7 Secure

7. Do you consider that the systems **support** your everyday activities at work?

Obstructive 1 2 3 4 5 6 7 Supportive

Dependability (continues)

Think of the systems you selected previously for answering the following questions.

8. Do you consider they are **predictable**? That is, do they always do what you think they will do? (e.g., remember that button you were unsure to click on because it wasn't clear what it would do? Do you remember a system that behaved in an inconsistent way? (e.g., performing operations over a dataset one way today and a different way the next day)

Unpredictable 1 2 3 4 5 6 7 Predictable

9. Do you think they are **accessible**? (e.g., color-blind people, hearing-problems people, and people with other kinds of disabilities can use them)

Inaccessible 1 2 3 4 5 6 7 Accessible

Dependability (continues)

Think of the systems you selected previously for answering the following questions.

10. Do you consider they are **findable**? (That is, it's easy to access the systems via a web link, an icon, and there is enough awareness of the systems' existence)

Not findable 1 2 3 4 5 6 7 Findable

11. Do you think they are easy to **control**? (e.g., graphic controls are easily manipulable, systems' parameters can be easily edited, dates can be entered easily, etc.)

Uncontrollable 1 2 3 4 5 6 7 Controllable

Perspicuity

Think of the systems you selected previously for answering the following questions.

12. How difficult is to **learn** how to use them?

Difficult to learn 1 2 3 4 5 6 7 Easy to learn

13. Do the systems convey information in a **clear** way? (e.g., through the use of lists, tables, or dialogs according to the information being presented)

Confusing 1 2 3 4 5 6 7 Clear

Perspicuity (continues)

Think of the systems you selected previously for answering the following questions.

14. Do you think they are **complex**? (i.e., the processes they execute look fuzzy to you or they convey information in a way that is more complicated than it should be)

Complex 1 2 3 4 5 6 7 Simple

15. Do they show **internal consistency**? (e.g., same iconography for the sames options across all windows, their main tools are always located in the same place, etc.)

Inconsistent 1 2 3 4 5 6 7 Consistent

16. Do they show **external consistency**? (e.g., use of standard commands like CTRL + P, CTRL + I, and CTRL + A for paste, print, and open instead of their own)

Inconsistent 1 2 3 4 5 6 7 Consistent

Efficiency

Think of the systems you selected previously for answering the following questions.

17. Are they **efficient**? (i.e., you can execute tasks as quickly as you'd like to execute them)

Inefficient 1 2 3 4 5 6 7 Efficient

18. Do they present data in an **organized** way? (e.g., the interfaces present information in a way that is easy to interpret)

Cluttered 1 2 3 4 5 6 7 Organized

Survey for Users

Hedonic Quality

(All these questions are related to the computer systems built in the Bank, particularly in the Central Bank Operating Systems Development Division **and that you have been in touch with**)

Please choose the number that is closer to how you feel about any particular characteristic.

Novelty

Think of the systems you selected previously for answering the following questions.

19. Do you think they are **innovative**? (e.g., they use new approaches for solving everyday challenges faced by your division)

Conservative 1 2 3 4 5 6 7 Innovative

20. Are they **modern**? (e.g., they present features such as progress saving, favorite options suggestions, GUI customization, team collaboration, pre-visualization, etc.)

Obsolete 1 2 3 4 5 6 7 Modern

Novelty

Think of the systems you selected previously for answering the following questions.

21. Do you think they are **creative**? (e.g. problem solving is done in an ingenious way that consumes less resources and it's equally effective as the traditional way)

Dull 1 2 3 4 5 6 7 Creative

22. Is the way they work **original**? (e.g. they make use of new algorithms that allow executing processes in a more efficient way compared to traditional ones, they make use of new graphical components that improve your interaction con the GUI, etc.)

Ordinary 1 2 3 4 5 6 7 Original

Stimulation

Think of the systems you selected previously for answering the following questions.

23. Do you think they add **value** to your everyday activities at work? (i.e., they allow you to complete crucial tasks that would normally take a valuable part of your time thus allowing your division to focus on improving its internal processes)

Inferior 1 2 3 4 5 6 7 Valuable

24. Do they generate **excitement** on you when using them? (e.g., surprise by how they execute tasks, strangeness, or happiness because you can achieve your work more quickly, etc.)

Boring 1 2 3 4 5 6 7 Exciting

Stimulation

Think of the systems you selected previously for answering the following questions.

25. Do you think they are **interesting**? (i.e., you feel curious about the way they interpret data and the way they communicate with other systems. You'd like to know more about how they perform certain tasks)

Not interesting 1 2 3 4 5 6 7 Interesting

26. Do you feel **motivated** when using them? (i.e., do the systems boost your everyday activities at work or they rather hinder them)

Demotivating 1 2 3 4 5 6 7 Motivating

Stimulation

Think of the systems you selected previously for answering the following questions.

27. Do you feel they are **exclusive**? (i.e., they have been designed to fit your division needs rather than everybody's needs in the General Directorate)

Standard 1 2 3 4 5 6 7 Exclusive

28. Do you feel **impressed** by them? (i.e., have you felt astonished by how easy they make look certain processes that if done manually would be a great burden)

Nondescript 1 2 3 4 5 6 7 Impressive

Survey for Users

Appeal

(All these questions are related to the computer systems built in the Bank, particularly in the Central Bank Operating Systems Development Division **and that you have been in touch with**)

Please choose the number that is closer to how you feel about any particular characteristic.

Think of the systems you selected previously for answering the following questions.

29. Are the systems **good** products?

Good 1 2 3 4 5 6 7 Bad

30. Do you **like** them? (i.e., are you satisfied with the features they have and the way they operate?)

Unlikable 1 2 3 4 5 6 7 Pleasing

31. Do you feel they are **attractive**? (i.e., they are inviting when you look at their graphical interfaces and the features they offer)

Unattractive 1 2 3 4 5 6 7 Attractive

Think of the systems you selected previously for answering the following questions.

32. Are they **friendly**? (i.e., they let you accomplish your activities with ease)

Unfriendly 1 2 3 4 5 6 7 Friendly

33. Do you **enjoy** using them? (i.e., do you feel pleasure when using them?)

Annoying 1 2 3 4 5 6 7 Enjoyable

34. Are they **aesthetically** pleasant? (i.e., do their interfaces have a design that is appealing to you?)

Unaesthetic 1 2 3 4 5 6 7 Aesthetic

Think of the systems you selected previously for answering the following questions.

35. Are they **desirable**? (i.e., if you didn't have them, would you advocate developing them?)

Undesirable 1 2 3 4 5 6 7 Desirable

Survey for Users

Open question

When answering the following questions, think of the development or implementation of future systems

37. Feel free to add any additional comments regarding the chosen systems category and the user experience those systems give you **OPTIONAL - TEXT**

Spanish Version

Encuesta: Evaluación de la Experiencia de Usuario (UX) de los Sistemas Operativos desarrollados en la DGOSP

El objetivo de esta encuesta es medir de manera cualitativa el nivel de Experiencia de Usuario (UX) de los sistemas de cómputo implementados en la Gerencia de Desarrollo de Sistemas Operativos (GDSO) de la Dirección de Apoyo a las Operaciones (DAO). Esta encuesta está diseñada para ser respondida por todos los analistas, jefes de oficina, subgerentes, gerentes y directores que hagan uso de dichos sistemas como parte de sus labores.

El diseño basado en UX es una nueva forma de construir productos, en este caso sistemas de cómputo que tomen en cuenta tanto aspectos ergonómicos (p.e.: eficacia, facilidad de uso, eficiencia) como aspectos de apreciación relacionados con el gusto mismo que genera el utilizarlos (p.e.: innovación, originalidad, estética). Ambos aspectos se combinan en la mente de los usuarios de dichos sistemas y les permite generar una evaluación de los mismos. Cualquier sistema provee una experiencia de usuario ya sea positiva o negativa independientemente de si este fue diseñado siguiendo principios de UX o no.

Esta encuesta forma parte de la tesis de Jonathan Laguna titulada "Desarrollo de una Política de Experiencia de Usuario (UX) para la Implementación de los Sistemas Operativos en el Banco de México". Actualmente, Jonathan está cursando su maestría en el departamento de Diseño de Sistemas y Administración del Instituto Tecnológico de Massachusetts (MIT). Antes de iniciar sus estudios en el MIT, Jonathan trabajó para la Gerencia de Desarrollo de Sistemas operativos por casi diez años como Analista de Desarrollo de Sistemas. Después de su graduación, Jonathan planea reincorporarse al Banco de México para seguir apoyando el desarrollo de los sistemas operativos desde una perspectiva más integral que considere el diseño basado en UX.

Esta encuesta es anónima y sólo Jonathan y su asesor de tesis, el Dr. Eric Rebentisch tendrán acceso a la información recabada. Tus respuestas ayudarán a detectar áreas de oportunidad para el mejoramiento de los sistemas operativos.

Duración estimada de la encuesta: 10 min.

Disponible hasta: Martes 30 de abril

Sección I. Datos Generales

Esta sección permitirá recabar datos generales para evaluar las respuestas de la encuesta desde diferentes ángulos.

1. ¿En qué área trabajas actualmente? *

1. Dirección de Operaciones Nacionales
2. Dirección de Operaciones Internacionales
3. Dirección de Apoyo a las Operaciones

2. ¿Cuál es tu antigüedad en el Banco? *

1. 0 a 10 años
2. Más de 10 años y hasta 20
3. Más de 20 años y hasta 30
4. Más de 30

3. ¿Qué carrera estudiaste? *

1. Economía
2. Actuaría
3. Contaduría
4. Finanzas
5. Informática
6. Matemáticas/Física
7. Sistemas Computacionales o afín
8. Otra (no es necesario especificar)

Sección II. Selección de Categoría de Sistemas de Cómputo

Las categorías agrupan sistemas operativos que se relacionan con el mismo negocio y/o que tienen interfaces gráficas similares con el fin de facilitar tus respuestas. No todos los sistemas fueron tomados en cuenta ya sea porque no tienen una interfaz gráfica, su uso es muy esporádico o se encuentran en proceso de liberación.

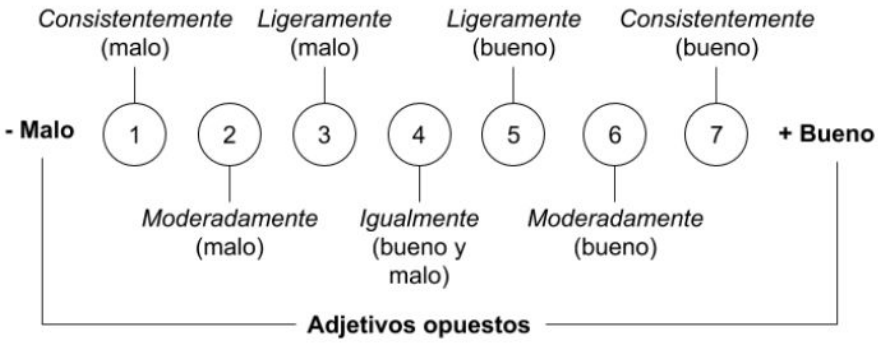
4. Piensa en los sistemas de cómputo que usas con mayor frecuencia para realizar tu trabajo y selecciona la categoría que mejor los describa. No necesariamente tienes que haber usado todos los sistemas incluidos en una categoría y quizás hayas usado algunos que no se encuentran en la que más te representa. NOTAS: a) las preguntas que siguen estarán basadas en esta selección, b) puedes volver a tomar la encuesta en caso de que quieras opinar sobre más de una categoría. *

1. POLITICA MONETARIA Y AGENCIA FINANCIERA "A"
2. POLITICA MONETARIA Y AGENCIA FINANCIERA "B"
3. POLITICA MONETARIA Y AGENCIA FINANCIERA "C"
4. CAMBIOS NACIONALES "A"
5. CAMBIOS NACIONALES "B"
6. GESTION DE OPERACIONES "A"
7. GESTION DE OPERACIONES "B"
8. GESTION DE OPERACIONES CON VALORES "A"
9. GESTION DE OPERACIONES CON VALORES "B"
10. ANALISIS DE MERCADOS NACIONALES "A"
11. ANALISIS DE MERCADOS NACIONALES "B"
12. PROGRAMACION FINANCIERA "A"
13. PROGRAMACION FINANCIERA "B"
14. OPERACIONES INTERNACIONALES "A"
15. OPERACIONES INTERNACIONALES "B"

Sección III. Ergonomía parte A - Piensa en los sistemas de la categoría que seleccionaste para responder las siguientes preguntas.

NOTA: Responde las preguntas seleccionando el número que más se acerque a tu postura. Los adjetivos ubicados en los extremos son opuestos. La izquierda siempre representará una connotación negativa y la derecha una positiva.

Interpretación de la escala del 1 al 7



A partir de aquí comienzan las preguntas de la sección III

Description (optional)

5. ¿Consideras que los sistemas cumplen con tus expectativas? (es decir, los sistemas hacen aquello para lo cual fueron creados y lo hacen de la forma en la que tú piensas deben hacerlo) *

	1	2	3	4	5	6	7	
No cumplen con expectativas	○	○	○	○	○	○	○	Cumplen con expectativas

6. ¿Son seguros? (Esta pregunta no se trata de seguridad ante ataques de Hackers, va enfocado a que las interfaces gráficas sean seguras de usar. Un ejemplo de sistema inseguro sería aquel en el que es muy fácil dar clic en el botón incorrecto provocando el borrado de información importante o la ejecución de una operación que no debía ser procesada.) *

	1	2	3	4	5	6	7	
Inseguros	○	○	○	○	○	○	○	Seguros

7. ¿Consideras que los sistemas facilitan tus actividades diarias en el trabajo? *

	1	2	3	4	5	6	7	
Dificultan	○	○	○	○	○	○	○	Facilitan

8. ¿Piensas que los sistemas son predecibles? (es decir, siempre hacen lo que tú piensas que harán. Un ejemplo de inconsistencia es cuando un sistema ejecuta operaciones sobre un mismo conjunto de datos de una forma un día y de otra forma al día siguiente.) *

	1	2	3	4	5	6	7	
Impredecibles	○	○	○	○	○	○	○	Predecibles

9. ¿Consideras que los sistemas son accesibles para cualquier persona? (incluidas personas que padecen daltonismo, es decir, ceguera a los colores verde, rojo, azul o amarillo, personas con problemas auditivos, y en general para personas con algún tipo de discapacidad) *

	1	2	3	4	5	6	7	
Inaccesibles	○	○	○	○	○	○	○	Accesibles

10. ¿Los sistemas son localizables? (es decir, es fácil acceder a ellos a través de un vínculo web, un ícono en el escritorio y todos los usuarios que podrían llegar a utilizar el sistema saben de su existencia) *

	1	2	3	4	5	6	7	
No localizables	○	○	○	○	○	○	○	Localizables

11. ¿Consideras que los sistemas son fáciles de controlar? (es decir, los controles gráficos son fácilmente manipulables, los parámetros del sistema pueden ser editados con facilidad, las fechas pueden ser ingresadas sin dificultad, etc.) *

	1	2	3	4	5	6	7	
Incontrolables	○	○	○	○	○	○	○	Controlables

Sección IV. Ergonomía parte B - Piensa en los sistemas de la categoría que seleccionaste para responder las siguientes preguntas.

Description (optional)

12. ¿Qué tan difícil es aprender a usarlos? *

	1	2	3	4	5	6	7	
Difícil de aprender	○	○	○	○	○	○	○	Fácil de aprender

13. ¿Consideras que los sistemas presentan la información de forma clara? (p.e.: a través de listas, tablas o mediante un cuadro de diálogo según el tipo de información a presentar) *

	1	2	3	4	5	6	7	
Confusa	○	○	○	○	○	○	○	Clara

14. ¿Piensas que los sistemas son complejos? (es decir, los procesos que ejecutan son una caja negra para ti) *

	1	2	3	4	5	6	7	
Complejos	○	○	○	○	○	○	○	Simples

15. ¿Consideras que los sistemas tienen consistencia interna? (Se refiere a la uniformidad de la interfaz, por ejemplo, que dentro de todas las ventanas de la aplicación se utilice la misma iconografía para las opciones que ejecutan el mismo proceso. En este caso, un botón cuyo fin sea imprimir siempre tendrá el mismo ícono sin importar en donde se encuentre el usuario dentro de la aplicación) *

	1	2	3	4	5	6	7	
Inconsistentes	○	○	○	○	○	○	○	Consistentes

16. ¿Consideras que los sistemas tienen consistencia externa? (Se refiere a si la aplicación es consistente con los estándares de desarrollo de software, por ejemplo, el uso de los comandos CTRL + P, CTRL + I, y CTRL + A para pegar, imprimir y abrir un archivo en lugar de sus propios comandos) *

	1	2	3	4	5	6	7	
Inconsistentes	○	○	○	○	○	○	○	Consistentes

17. ¿Los sistemas son eficientes? (Esta pregunta sólo considera la eficiencia en el manejo de las interfaces gráficas y no en aspectos como la lentitud de la red o el servidor. Un ejemplo es que quizás un formulario te pide capturar campos que son innecesarios o quizás algunos campos no son fáciles de ingresar como en el caso de las fechas o los montos, alentando así la ejecución de tareas.) *

	1	2	3	4	5	6	7	
Ineficientes	○	○	○	○	○	○	○	Eficientes

18. ¿Las interfaces de los sistemas lucen saturadas? (p.e.: la interfaz está tan saturada de controles e información que se vuelve difícil interpretarla) *

	1	2	3	4	5	6	7	
Saturadas	○	○	○	○	○	○	○	Balanceadas

Sección V. Apreciación - Piensa en los sistemas de la categoría que seleccionaste anteriormente para responder las siguientes preguntas.

Description (optional)

19. ¿Los sistemas son innovadores? (p.e.: utilizan nuevos métodos para resolver los problemas que enfrenta tu área) *

	1	2	3	4	5	6	7	
Conservadores	○	○	○	○	○	○	○	Innovadores

20. ¿Consideras que son modernos? (p.e.: permiten compartir información con otros usuarios de manera ágil, previenen tareas repetitivas como guardar tus avances, detectan cuáles son las opciones más utilizadas y las sugieren, permiten personalizar la vista, colaborar con tus compañeros de trabajo, pre-visualizar los resultados de una operación, etc.) *

	1	2	3	4	5	6	7	
Obsoletos	○	○	○	○	○	○	○	Modernos

21. ¿Piensas que los sistemas son creativos? (p.e.: resuelven los problemas de una manera que es ingeniosa, utiliza menos recursos y es igual de efectiva que la forma tradicional) *

	1	2	3	4	5	6	7	
No creativos	○	○	○	○	○	○	○	Creativos

22. ¿Son originales? (p.e.: utilizan nuevos algoritmos que permiten ejecutar procesos de forma más eficiente comparado con los algoritmos tradicionalmente utilizados en tu área, utilizan nuevos controles gráficos que mejoran tu interacción con la interfaz, etc.) *

	1	2	3	4	5	6	7	
Ordinarios	○	○	○	○	○	○	○	Originales

23. ¿Consideras que los sistemas son valiosos para tu área? (para responder esta pregunta piensa en las afectaciones que tu área sufriría si estos sistemas no existieran) *

	1	2	3	4	5	6	7	
No valiosos	○	○	○	○	○	○	○	Valiosos

24. ¿Te generan algún tipo de emoción? (p.e.: sorpresa al ver cómo ejecutan alguna tarea, extrañeza, felicidad por lo rápido que te permiten hacer tu trabajo, etc.) *

	1	2	3	4	5	6	7	
No emocionantes	○	○	○	○	○	○	○	Emocionantes

25. ¿Consideras que los sistemas son interesantes? (p.e.: su uso te genera ideas de mejoras o nuevas funcionalidades que podrían agregársele, o sientes curiosidad por ver si podrían conectarse con otros sistemas para agilizar la operación, etc.) *

	1	2	3	4	5	6	7	
No interesantes	○	○	○	○	○	○	○	Interesantes

26. ¿Te motivan a realizar tu trabajo? (es decir, mejoran la forma en la que los procesos de tu área son ejecutados en vez de ralentizarlos o hacerlos tediosos) *

	1	2	3	4	5	6	7	
Te desmotivan	○	○	○	○	○	○	○	Te motivan

27. ¿Sientes que los sistemas son exclusivos? (es decir, están diseñados a la medida de tu área) *

	1	2	3	4	5	6	7	
Estándar	○	○	○	○	○	○	○	Exclusivos

28. ¿Te has sentido impresionado al usarlos? (p.e.: los sistemas hacen ver sencillas tareas que era muy tardado o difícil ejecutar a mano) *

	1	2	3	4	5	6	7	
Indiferente	○	○	○	○	○	○	○	Impresionado

Sección VI. Evaluación global - Piensa en los sistemas de la categoría que seleccionaste para responder las siguientes preguntas.

Description (optional)

29. ¿Consideras que los sistemas son buenos o malos en su calidad de producto? *

	1	2	3	4	5	6	7	
Malos	○	○	○	○	○	○	○	Buenos

30. ¿Te gustan? (es decir, estás satisfecho con sus características y la manera en la que operan) *

	1	2	3	4	5	6	7	
No me gustan	○	○	○	○	○	○	○	Me gustan

31. ¿Consideras que son atractivos? (es decir, te sientes atraído por sus interfaces gráficas y las prestaciones que ofrecen) *

	1	2	3	4	5	6	7	
No atractivos	○	○	○	○	○	○	○	Atractivos

32. ¿Son amigables? (es decir, te permiten cumplir con tus actividades de una manera sencilla) *

	1	2	3	4	5	6	7	
No amigables	○	○	○	○	○	○	○	Amigables

33. ¿Disfrutas usándolos? (es decir, sus cualidades te generan una sensación de bienestar) *

	1	2	3	4	5	6	7	
No disfruto	○	○	○	○	○	○	○	Disfruto

34. ¿Son estéticos? (p.e.: sus interfaces tienen un diseño agradable a la vista) *

	1	2	3	4	5	6	7	
No estéticos	○	○	○	○	○	○	○	Estéticos

35. ¿Consideras que son deseables? (es decir, si no existieran, solicitarías que fueran desarrollados) *

	1	2	3	4	5	6	7	
No deseables	○	○	○	○	○	○	○	Deseables

Sección VII. Comentarios

Esta es la última sección de esta encuesta

36. Siéntete libre de agregar cualquier comentario relacionado con los sistemas que seleccionaste anteriormente y la experiencia de usuario que estos te han brindado al usarlos.

Long answer text

Appendix B. Survey for Developers

English Version

Survey for Developers Structure

- **Landing page**
- **Demographics** (3 questions)
- **Computer Systems Category Selection** (1 question)
- **Ergonomic Quality** (14 questions divided in two sections)
- **Hedonic Quality** (10 questions)
- **Appeal** (7 questions)
- **Open question** (1 question)

36 questions in total

If every question is answered in 15 secs (or 4 questions per minute), this survey should not take more than 10 min approx.

Survey for Developers

The goal of this survey is to measure the current User Experience (UX) level of computer systems that have been developed by the *Operating Systems Development Division (OSD-D)* within the Directorate of Operations Support (DOS) from the developers' perspective. This survey has been designed to be responded by all analysts, specialty leaders, deputy managers, and manager within the OSD-D.
UX design is a new way of building systems that account for both *ergonomic* (e.g., effectiveness, learnability, efficiency) and *hedonic* (e.g., innovativeness, originality, aesthetics) aspects of software. Both aspects combine in Developers' minds and allow them to come up with a general appeal of systems. Any computer system delivers a user experience either positive or negative, regardless of whether UX design was taken into account or not.

This survey is part of Jonathan Laguna's thesis titled *Implementing a User Experience Policy for Operating Computer Systems Development in the Central Bank of Mexico*. Jonathan is currently enrolled in the System Design in Management Program (SDM) at MIT. He worked for the Central Bank of Mexico in the CBOS division for almost ten years as a systems development analyst before coming to MIT, and he is planning to continue working for the Bank after graduation.

This survey is intended to be anonymous. Only Jonathan and his thesis supervisor Dr. Eric Rebentisch will have access to the information here collected. Your responses will help spot opportunity areas for the improvement of operating systems development.

Survey for Developers

Demographics

1. Select the number of years you've been working for the Bank **SINGLE CHOICE**
    a. 0 to 10
    b. More than 10 and up to 20
    c. More than 20 and up to 30
    d. More than 30

2. Select your background **SINGLE CHOICE**

    a. Computer Systems
    a. Informatics
    b. Mathematics/Physics
    c. Other (don't have to specify)

3. How familiar are you with the term UX? **SINGLE CHOICE**
    a. Never heard of it
    b. Heard of it but not familiar
    c. I know it and I have applied some basic concepts to the systems development
    d. I have taken courses about UX and I have applied them to the systems development

Survey for Developers

Computer Systems Category Selection

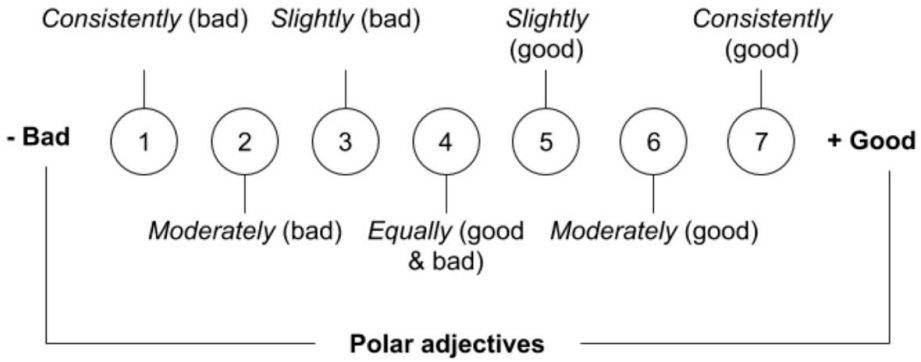
The categories group systems that relate on business and/or that have similar GUIs so you can answer questions more easily. Not all systems were included either because they don't have a GUI, they are used sporadically, or they are under a roll out process.

4. Please select the category that better describes the systems you're in charge of at work. It doesn't matter if the category doesn't cover all the systems you work on regularly. NOTES: a) the rest of the survey will touch upon the selected category, b) you can take this survey again in case you want to give your opinion for another category. **SINGLE CHOICE**
    a. Monetary Policy A (*names of systems in this category*)
    b. Monetary Policy B (*names of systems in this category*)
    c. Monetary Policy C (*names of systems in this category*)
    d. Domestic Exchange Rates A (*names of systems in this category)*
    e. Domestic Exchange Rates B (*names of systems in this category)*
    f. Operations Management A (*names of systems in this category*)
    g. Operations Management B (*names of systems in this category*)
    h. Securities Operations Management A (*names of systems in this category*)
    i. Securities Operations Management B (*names of systems in this category*)
    j. Domestic Markets Analysis A (*names of systems in this category*)
    k. Domestic Markets Analysis B (*names of systems in this category*)
    l. Financial Programming A (*names of systems in this category*)
    m. Financial Programming B (*names of systems in this category*)
    n. International Operations Management A (*names of systems in this category*)
    o. International Operations Management B (*names of systems in this category*)

Survey for Developers

Ergonomic Quality

(Think of the systems in the category you selected previously to answer the following questions)
NOTE: Answer the questions selecting the number that is closer to your posture. The adjectives located in both extremes are polar. The left will always represent a negative connotation whereas the right a positive one.



Survey for Developers

Ergonomic Quality

(All these questions are related to the computer systems built in the Bank, particularly in the Central Bank Operating Systems Development Division **and that you have been in touch with**)

Please choose the number that is closer to how you feel about any particular characteristic.

Dependability

Think of the systems you selected previously for answering the following questions.

5. Do you consider they **meet** users' expectations? (i.e., they do what they were expected to do and they do it the way users would like them to do it)

Meet expectations 1 2 3 4 5 6 7 Doesn't meet expectations

Survey for Developers

Ergonomic Quality

(All these questions are related to the computer systems built in the Bank, particularly in the Central Bank Operating Systems Development Division **and that you have been in touch with**)

Please choose the number that is closer to how you feel about any particular characteristic.

Dependability (continues)

Think of the systems you selected previously for answering the following questions.

6. Do you consider they are **secure** for users? (This is not about security against hackers but rather on their graphical user interfaces use. For example, an insecure system would be one where it's very easy to click on the wrong button causing information deletion or the execution of an operation that didn't have to be processed at all)

Not secure 1 2 3 4 5 6 7 Secure

7. Do you consider that the systems **support** users' everyday activities at work?

Obstructive 1 2 3 4 5 6 7 Supportive

Dependability (continues)

Think of the systems you selected previously for answering the following questions.

8. Do you consider they are **predictable** for users? That is, do they always do what users think they will do? (e.g., remember that button you were unsure to click on because it wasn't clear what it would do? Do you remember a system that behaved in an inconsistent way such as performing operations over a dataset one way today and a different way the next day)

Unpredictable 1 2 3 4 5 6 7 Predictable

9. Do you think they are **accessible** for users? (e.g., daltonic or color-blind users that have trouble distinguishing red, green, yellow, and blue, hearing-problems people, and people with other kinds of disabilities can use them)

Inaccessible 1 2 3 4 5 6 7 Accessible

Dependability (continues)

Think of the systems you selected previously for answering the following questions.

10. Do you consider they are **findable** for users? (That is, it's easy for users to access systems via a web link, an icon, and there is enough awareness of the systems' existence)

Not findable 1 2 3 4 5 6 7 Findable

11. Do you think they are easy to **control** for users? (e.g., users can easily manipulate graphic controls, edit systems' parameters, enter dates, etc.)

Uncontrollable 1 2 3 4 5 6 7 Controllable

Perspicuity

Think of the systems you selected previously for answering the following questions.

12. How difficult is for users to **learn** how to use them?

Difficult to learn 1 2 3 4 5 6 7 Easy to learn

13. Do the systems convey information in a **clear** way for users? (e.g., through the use of lists, tables, or dialogs according to the information being presented to them)

Confusing 1 2 3 4 5 6 7 Clear

Perspicuity (continues)

Think of the systems you selected previously for answering the following questions.

14. Do you think users see them as **complex**? (i.e., the processes they execute look fuzzy to users or they convey information in a way that is more complicated than it should be)

Complex 1 2 3 4 5 6 7 Simple

15. Do you think users see them as **internal consistent**? (e.g., same iconography for the sames options across all windows, their main tools are always located in the same place, etc.)

Inconsistent 1 2 3 4 5 6 7 Consistent

16. Do you think users see them as **external consistent**? (e.g., use of standard commands like CTRL + P, CTRL + I, and CTRL + A for paste, print, and open instead of their own)

Inconsistent 1 2 3 4 5 6 7 Consistent

Efficiency

Think of the systems you selected previously for answering the following questions.

17. Do you think they **efficient** to users? (i.e., users can execute tasks as quickly as they'd like to execute them)

Inefficient 1 2 3 4 5 6 7 Efficient

18. Do they present data in an **organized** way to users? (e.g., the interfaces present information in a way that is easy to interpret for users)

Cluttered 1 2 3 4 5 6 7 Organized

Survey for Developers

Hedonic Quality

(All these questions are related to the computer systems built in the Bank, particularly in the Central Bank Operating Systems Development Division **and that you have been in touch with**)

Please choose the number that is closer to how you feel about any particular characteristic.

Novelty

Think of the systems you selected previously for answering the following questions.

19. Do you think users see them as **innovative**? (e.g., they use new approaches for solving everyday challenges faced by users' divisions)

Conservative 1 2 3 4 5 6 7 Innovative

20. Do you think they look **modern** to users? (e.g., they present features such as progress saving, favorite options suggestions, GUI customization, team collaboration, pre-visualization, etc.)

Obsolete 1 2 3 4 5 6 7 Modern

Novelty

Think of the systems you selected previously for answering the following questions.

21. Do you think users see them as **creative**? (e.g. problem solving is done in an ingenious way that consumes less resources and it's equally effective as the traditional way)

Dull 1 2 3 4 5 6 7 Creative

22. Do you feel like users see the way they work as **original**? (e.g. they make use of new algorithms that allow executing processes in a more efficient way compared to traditional ones, they make use of new graphical components that improve your interaction con the GUI, etc.)

Ordinary 1 2 3 4 5 6 7 Original

Stimulation

Think of the systems you selected previously for answering the following questions.

23. Do you think they add **value** to users' everyday activities at work? (i.e., they allow users to complete crucial tasks that would normally take a valuable part of their time thus allowing their divisions to focus on improving their internal processes)

Inferior 1 2 3 4 5 6 7 Valuable

24. Do you think they generate **excitement** on users? (e.g., surprise by how systems execute tasks, strangeness, or happiness because users can achieve their work more quickly, etc.)

Boring 1 2 3 4 5 6 7 Exciting

Stimulation

Think of the systems you selected previously for answering the following questions.

25. Do you think they are **interesting** to users? (i.e., users feel curious about the way systems interpret data and the way their systems communicate with other systems. Additionally, users would like to know more about how systems perform certain tasks)

Not interesting 1 2 3 4 5 6 7 Interesting

26. Do you think users feel **motivated** when using them? (i.e., do the systems boost users' everyday activities at work or they rather hinder them)

Demotivating 1 2 3 4 5 6 7 Motivating

Stimulation

Think of the systems you selected previously for answering the following questions.

27. Do you think they feel **exclusive** to users? (i.e., they have been designed specifically to fit users' divisions needs)

Standard 1 2 3 4 5 6 7 Exclusive

28. Do you think users feel **impressed** by them? (i.e., users have felt astonished by how easy systems make look certain processes that if done manually would be a great burden)

Nondescript 1 2 3 4 5 6 7 Impressive

Survey for Developers

Appeal

(All these questions are related to the computer systems built in the Bank, particularly in the Central Bank Operating Systems Development Division **and that you have been in touch with**)

Please choose the number that is closer to how you feel about any particular characteristic.

Think of the systems you selected previously for answering the following questions.

29. Do you consider that users see them as **good** products?

Good 1 2 3 4 5 6 7 Bad

30. Do you think users **like** them? (i.e., users are satisfied with the features systems have and the way they operate?)

Unlikable 1 2 3 4 5 6 7 Pleasing

31. Do you feel they are **attractive** to users? (i.e., they are inviting when users look at their graphical interfaces and the features they offer)

Unattractive 1 2 3 4 5 6 7 Attractive

Think of the systems you selected previously for answering the following questions.

32. Are they **friendly** to users? (i.e., they let users accomplish their activities with ease)

Unfriendly 1 2 3 4 5 6 7 Friendly

33. Do you think users **enjoy** using them? (i.e., users feel pleasure when using them)

Annoying 1 2 3 4 5 6 7 Enjoyable

34. Do you think they are **aesthetically** pleasant for users? (i.e., their interfaces have a design that is appealing to users)

Unaesthetic 1 2 3 4 5 6 7 Aesthetic

Think of the systems you selected previously for answering the following questions.

35. Are they **desirable** for users? (i.e., if users didn't have them, would they advocate developing them?)

Undesirable 1 2 3 4 5 6 7 Desirable

Spanish Version

Encuesta: Evaluación de la Experiencia de Usuario (UX) de los Sistemas Operativos desarrollados en la GDSO.

El objetivo de esta encuesta es medir de manera cualitativa el nivel de Experiencia de Usuario (UX) de los sistemas de cómputo implementados en la Gerencia de Desarrollo de Sistemas Operativos (GDSO) de la Dirección de Apoyo a las Operaciones (DAO). Esta encuesta está diseñada para ser respondida por todos los analistas, líderes de especialidad, subgerentes, y gerente que se encargan del desarrollo de estos sistemas.

El diseño basado en UX es una nueva forma de construir productos, en este caso sistemas de cómputo que tomen en cuenta tanto aspectos ergonómicos (p.e.: eficacia, facilidad de uso, eficiencia) como aspectos de apreciación relacionados con el gusto mismo que genera el utilizarlos (p.e.: innovación, originalidad, estética). Ambos aspectos se combinan en la mente de los usuarios de dichos sistemas y les permite generar una evaluación de los mismos. Cualquier sistema provee una experiencia de usuario ya sea positiva o negativa independientemente de si este fue diseñado siguiendo principios de UX o no.

Esta encuesta forma parte de la tesis de Jonathan Laguna titulada "Desarrollo de una Política de Experiencia de Usuario (UX) para la Implementación de los Sistemas Operativos en el Banco de México". Actualmente, Jonathan está cursando su maestría en el departamento de Diseño de Sistemas y Administración del Instituto Tecnológico de Massachusetts (MIT). Antes de iniciar sus estudios en el MIT, Jonathan trabajó para la Gerencia de Desarrollo de Sistemas operativos por casi diez años como Analista de Desarrollo de Sistemas. Después de su graduación, Jonathan planea reincorporarse al Banco de México para seguir apoyando el desarrollo de los sistemas operativos desde una perspectiva más integral que considere el diseño basado en UX.

Esta encuesta es anónima y sólo Jonathan y su asesor de tesis, el Dr. Eric Rebentisch tendrán acceso a la información recabada. Tus respuestas ayudarán a contrastar la perspectiva que tiene el área de desarrollo respecto a UX con la perspectiva real que tienen los usuarios obtenida por medio de una encuesta paralela y permitirá detectar áreas de oportunidad para el mejoramiento de los sistemas operativos.

Duración estimada de la encuesta: 10 min.

Disponible hasta: Martes 30 de abril

Sección I. Datos Generales

Esta sección permitirá recabar datos generales para evaluar las respuestas de la encuesta desde diferentes ángulos.

1. ¿Cuál es tu antigüedad en el Banco? *

1. 0 a 10 años
2. Más de 10 años y hasta 20
3. Más de 20 años y hasta 30
4. Más de 30

2. ¿Qué carrera estudiaste? *

1. Informática
2. Matemáticas/Física
3. Sistemas Computacionales
4. Otra (no es necesario especificar)

3. ¿Qué tan familiarizado estás con el término Experiencia de Usuario/User Experience/UX? *

- ◯ Nunca lo había escuchado
- ◯ Lo he escuchado pero no estoy muy familiarizado
- ◯ Lo conozco e incluso he aplicado algunos conceptos básicos de UX en el desarrollo de sistemas
- ◯ He tomado cursos de UX y los he aplicado en el desarrollo de sistemas

Sección II. Selección de Categoría de Sistemas de Cómputo

Las categorías agrupan sistemas operativos que se relacionan con el mismo negocio y/o que tienen interfaces gráficas similares con el fin de facilitar tus respuestas. No todos los sistemas fueron tomados en cuenta ya sea porque no tienen una interfaz gráfica, su uso es muy esporádico o se encuentran en proceso de liberación.

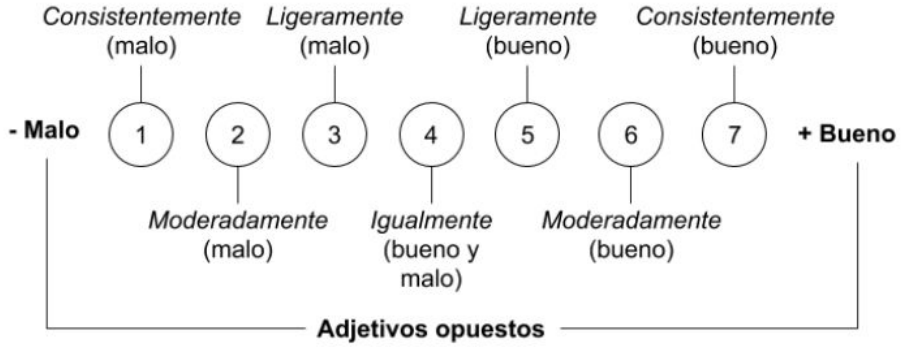
4. Piensa en los sistemas de cómputo a los que estás asignado y selecciona la categoría que mejor los describa. No necesariamente tienes que haber participado en el desarrollo de todos los sistemas incluidos en una categoría y quizás hayas estado en contacto con algunos que no se encuentran en la que más te representa. NOTAS: a) las preguntas que siguen estarán basadas en esta selección, b) puedes volver a tomar la encuesta en caso de que quieras opinar sobre más de una categoría. *

1. POLITICA MONETARIA Y AGENCIA FINANCIERA "A"
2. POLITICA MONETARIA Y AGENCIA FINANCIERA "B"
3. POLITICA MONETARIA Y AGENCIA FINANCIERA "C"
4. CAMBIOS NACIONALES "A"
5. CAMBIOS NACIONALES "B"
6. GESTION DE OPERACIONES "A"
7. GESTION DE OPERACIONES "B"
8. GESTION DE OPERACIONES CON VALORES "A"
9. GESTION DE OPERACIONES CON VALORES "B"
10. ANALISIS DE MERCADOS NACIONALES "A"
11. ANALISIS DE MERCADOS NACIONALES "B"
12. PROGRAMACION FINANCIERA "A"
13. PROGRAMACION FINANCIERA "B"
14. OPERACIONES INTERNACIONALES "A"
15. OPERACIONES INTERNACIONALES "B"

Sección III. Ergonomía parte A - Piensa en los sistemas de la categoría que seleccionaste para responder las siguientes preguntas.

NOTA: Responde las preguntas seleccionando el número que más se acerque a tu postura. Los adjetivos ubicados en los extremos son opuestos. La izquierda siempre representará una connotación negativa y la derecha una positiva.

Interpretación de la escala del 1 al 7



A partir de aquí comienzan las preguntas de la sección III

Description (optional)

5. ¿Consideras que los sistemas cumplen con las expectativas de los usuarios? (es decir, los sistemas hacen aquello para lo cual fueron creados y lo hacen de la forma en la que los usuarios consideran debería hacerse) *

	1	2	3	4	5	6	7	
No cumplen con expectativas	○	○	○	○	○	○	○	Cumplen con expectativas

6. ¿Consideras que sus interfaces gráficas son seguras de usar para los usuarios? (Un ejemplo de sistema inseguro sería aquel en el que es muy fácil dar clic en el botón incorrecto provocando el borrado de información importante o la ejecución de una operación que no debía ser procesada) *

	1	2	3	4	5	6	7	
Inseguras	○	○	○	○	○	○	○	Seguras

7. ¿Consideras que los sistemas facilitan el trabajo de los usuarios? *

	1	2	3	4	5	6	7	
Dificultan	○	○	○	○	○	○	○	Facilitan

8. ¿Piensas que los sistemas son predecibles para los usuarios? (es decir, siempre hacen lo que los usuarios piensan que harán o a veces estos son sorprendidos por comportamientos inesperados como por ejemplo incongruencia en el manejo de los decimales) *

	1	2	3	4	5	6	7	
Impredecibles	○	○	○	○	○	○	○	Predecibles

9. ¿Consideras que los sistemas son accesibles para cualquier usuario? (incluidos aquellos que pudieran padecer daltonismo, es decir, ceguera a los colores verde, rojo, azul o amarillo, usuarios con problemas auditivos, y en general usuarios con algún tipo de discapacidad) *

	1	2	3	4	5	6	7	
Inaccesibles	○	○	○	○	○	○	○	Accesibles

10. ¿Los sistemas son localizables para los usuarios? (es decir, es fácil acceder a ellos a través de un vínculo web, un ícono en el escritorio y todos los usuarios que podrían llegar a utilizar el sistema saben de su existencia) *

	1	2	3	4	5	6	7	
No localizables	○	○	○	○	○	○	○	Localizables

11. ¿Consideras que los usuarios pueden controlar los sistemas? (es decir, los controles gráficos son fácilmente manipulables, los parámetros del sistema pueden ser editados con facilidad, las fechas pueden ser ingresadas sin dificultad, etc.) *

	1	2	3	4	5	6	7	
Incontrolables	◯	◯	◯	◯	◯	◯	◯	Controlables

Sección IV. Ergonomía parte B - Piensa en los sistemas de la categoría que seleccionaste para responder las siguientes preguntas.

Description (optional)

12. ¿Qué tan difícil consideras es para los usuarios aprender a usarlos? *

	1	2	3	4	5	6	7	
Difícil de aprender	◯	◯	◯	◯	◯	◯	◯	Fácil de aprender

13. ¿Consideras que los sistemas presentan la información de forma clara a los usuarios? (p.e.: a través de listas, tablas o mediante un cuadro de diálogo según el tipo de información a presentar) *

	1	2	3	4	5	6	7	
Confusa	◯	◯	◯	◯	◯	◯	◯	Clara

14. ¿Piensas que los sistemas son complejos para los usuarios? (por ejemplo, los procesos que ejecutan son una caja negra para ellos) *

	1	2	3	4	5	6	7	
Complejos	◯	◯	◯	◯	◯	◯	◯	Simples

15. ¿Consideras que los sistemas tienen consistencia INTERNA para los usuarios? (Se refiere a la uniformidad de la interfaz, por ejemplo, que dentro de todas las ventanas de la aplicación se utilice la misma iconografía para las opciones que ejecutan el mismo proceso. En este caso, un botón cuyo fin sea imprimir siempre tendrá el mismo ícono sin importar en donde se encuentre el usuario dentro de la aplicación) *

	1	2	3	4	5	6	7	
Inconsistentes	◯	◯	◯	◯	◯	◯	◯	Consistentes

16. ¿Consideras que los sistemas tienen consistencia EXTERNA para los usuarios? (Se refiere a si la aplicación es consistente con los estándares de desarrollo de software, por ejemplo, el uso de los comandos CTRL + P, CTRL + I, y CTRL + A para pegar, imprimir y abrir un archivo en lugar de sus propios comandos) *

	1	2	3	4	5	6	7	
Inconsistentes	○	○	○	○	○	○	○	Consistentes

17. ¿Piensas que los sistemas son eficientes desde la perspectiva de los usuarios? (Esta pregunta sólo considera la eficiencia en el manejo de las interfaces gráficas y no en aspectos como la lentitud de la red o el servidor. Un ejemplo es que quizás un formulario pide al usuario capturar campos que son innecesarios o quizás algunos campos no son fáciles de ingresar como en el caso de las fechas o los montos, alentando así la ejecución de tareas.) *

	1	2	3	4	5	6	7	
Ineficientes	○	○	○	○	○	○	○	Eficientes

18. ¿Las interfaces de los sistemas lucen saturadas para los usuarios? (p.e.: la interfaz está tan saturada de controles e información que a los usuarios se les hace difícil interpretarla) *

	1	2	3	4	5	6	7	
Saturadas	○	○	○	○	○	○	○	Balanceadas

Sección V. Apreciación - Piensa en los sistemas de la categoría que seleccionaste anteriormente para responder las siguientes preguntas.

Description (optional)

19. ¿Consideras que los sistemas son innovadores para los usuarios? (p.e.: utilizan nuevos métodos para resolver los problemas que enfrentan en sus áreas) *

	1	2	3	4	5	6	7	
Conservadores	○	○	○	○	○	○	○	Innovadores

20. ¿Consideras que los sistemas son modernos para los usuarios? (p.e.: permiten compartir información con otros usuarios de manera ágil, previenen tareas repetitivas como guardar sus avances, detectan cuáles son las opciones más utilizadas y las sugieren, permiten personalizar la vista, colaborar con sus compañeros de trabajo, pre-visualizar los resultados de una operación, etc.) *

	1	2	3	4	5	6	7	
Obsoletos	○	○	○	○	○	○	○	Modernos

21. ¿Piensas que los sistemas son creativos desde la perspectiva de los usuarios? (p.e.: resuelven los problemas de una manera que es ingeniosa, utiliza menos recursos y es igual de efectiva que la forma tradicional) *

	1	2	3	4	5	6	7	
No creativos	○	○	○	○	○	○	○	Creativos

22. ¿Consideras que son originales para los usuarios? (p.e.: utilizan nuevos algoritmos que permiten ejecutar procesos de forma más eficiente comparado con los algoritmos tradicionalmente utilizados en las áreas usuarias, utilizan nuevos controles gráficos que mejoran la interacción de los usuarios con la interfaz, etc.) *

	1	2	3	4	5	6	7	
Ordinarios	○	○	○	○	○	○	○	Originales

23. ¿Consideras que los sistemas son valiosos para los usuarios? (para responder esta pregunta piensa en las afectaciones que los usuarios podrían tener si estos sistemas no existieran) *

	1	2	3	4	5	6	7	
No valiosos	○	○	○	○	○	○	○	Valiosos

24. ¿Consideras que generan algún tipo de emoción a los usuarios? (p.e.: sorpresa al ver cómo ejecutan alguna tarea, extrañeza, felicidad por lo rápido que les permiten hacer su trabajo, etc.) *

	1	2	3	4	5	6	7	
No emocionantes	○	○	○	○	○	○	○	Emocionantes

25. ¿Consideras que los sistemas son interesantes para los usuarios? (p.e.: su uso les genera ideas de mejoras o nuevas funcionalidades que podrían agregársele, o sienten curiosidad por ver si podrían conectarse con otros sistemas para agilizar la operación, etc.) *

	1	2	3	4	5	6	7	
No interesantes	○	○	○	○	○	○	○	Interesantes

26. ¿Piensas que motivan a los usuarios a realizar su trabajo? (es decir, mejoran la forma en la que los procesos de las áreas usuarias son ejecutados en vez de ralentizarlos o hacerlos tediosos) *

	1	2	3	4	5	6	7	
Desmotivan	○	○	○	○	○	○	○	Motivan

27. ¿Sientes que los sistemas son exclusivos para las áreas usuarias? (es decir, están diseñados a su medida) *

	1	2	3	4	5	6	7	
Estándar	○	○	○	○	○	○	○	Exclusivos

28. ¿Piensas que los usuarios se han sentido impresionados al usarlos? (p.e.: los sistemas hacen ver sencillas tareas que era muy tardado o difícil ejecutar a mano) *

	1	2	3	4	5	6	7	
Indiferentes	○	○	○	○	○	○	○	Impresionados

Sección VI. Evaluación global - Piensa en los sistemas de la categoría que seleccionaste para responder las siguientes preguntas.

Description (optional)

29. ¿Consideras que los sistemas son buenos o malos en su calidad de producto para los usuarios? *

	1	2	3	4	5	6	7	
Malos	◯	◯	◯	◯	◯	◯	◯	Buenos

30. ¿Crees que a los usuarios les gustan? (es decir, están satisfechos con sus características y la manera en la que operan) *

	1	2	3	4	5	6	7	
No les gustan	◯	◯	◯	◯	◯	◯	◯	Les gustan

31. ¿Consideras que son atractivos para los usuarios? (es decir, se sienten atraídos por sus interfaces gráficas y las prestaciones que ofrecen) *

	1	2	3	4	5	6	7	
No atractivos	◯	◯	◯	◯	◯	◯	◯	Atractivos

32. ¿Consideras que los usuarios los ven como amigables? (es decir, les permiten cumplir con sus actividades de una manera sencilla) *

	1	2	3	4	5	6	7	
No amigables	◯	◯	◯	◯	◯	◯	◯	Amigables

33. ¿Crees que los usuarios disfrutan usándolos? (es decir, sus cualidades les generan una sensación de bienestar) *

	1	2	3	4	5	6	7	
No disfrutan	○	○	○	○	○	○	○	Disfrutan

34. ¿Piensas que los usuarios los consideran estéticos? (p.e.: sus interfaces tienen un diseño agradable a la vista) *

	1	2	3	4	5	6	7	
No estéticos	○	○	○	○	○	○	○	Estéticos

35. ¿Consideras que son deseables para los usuarios? (es decir, si no existieran, los usuarios solicitarían que fueran desarrollados) *

	1	2	3	4	5	6	7	
No deseables	○	○	○	○	○	○	○	Deseables

Sección VII. Comentarios

Esta es la última sección de esta encuesta

36. Siéntete libre de agregar cualquier comentario relacionado con los sistemas que seleccionaste anteriormente y la experiencia de usuario que consideras estos han brindado.

Long answer text