

Whitepaper

User Experience Testing

Embracing the User Interface and the User Experience
as Core Business Values



sqs.com

Author: Sandra Harrison
Consultant
SQS Group Limited
United Kingdom

Published: August 2013



SANDRA HARRISON

Consultant

sandra.harrison@sqs.com

With 15 years' experience in a Belfast design company, Sandra graduated in May 2010 with a BSc (Hons) in Interactive Multimedia Design. Since beginning her career with SQS in September 2010 she has discovered a passion for user interface (UI) and user experience (UX) based testing, and the values that they can create for business in a world where users have grown up with technology and are now largely defined by their relationship to it. She believes that superior UI and UX should be embraced as core business values, and that a product should compete on quality and not just features alone. Her time with SQS has enabled her to utilise her background in design alongside her skills in test analysis to carve out her UI & UX specialisms across the Utilities, Education and Insurance sectors.

Contents

1.	Management Summary	4
2.	Introduction	5
3.	Market – Current Status and Outlook	7
4.	Why UX is Good for Business	8
	4.1. Acquiring Support for UX	8
	4.2. Example: Online Banking	9
	4.3. Business Cases for Improved UX	10
5.	Tools for Testing the UI and UX.	11
	5.1. Getting to Know the User.	11
	5.2. Wireframes and Screen Compositions	11
	5.3. Style Guides.	13
6.	Testing the UI	14
	6.1. UX Considerations when Functional Testing	14
	6.2. Features and Functionality	14
	6.3. Screen Layout	14
	6.4. Experience Test Cases.	15
	6.5. Error Handling	16
	6.6. UI Tests	16
7.	Conclusion and Outlook	19
8.	Bibliographical References	20

1. Management Summary

In recent years there have been major developments in the software marketplace that are bringing better and more usable software directly to consumers, giving them an insight into what is now possible with newly built software applications. This is raising expectations and users have become much less tolerant of bad design and poorly thought out user interface (UI) and user experience (UX) – especially when the application has been delivered to help them do their jobs and complete their daily tasks in the workplace.

Because we are all now consumers of technology, and as such know what technology is capable of, what is available to us in the workplace is often in stark contrast. But, in fact, in most cases a few polishing touches and a little attention to ‘look & feel’ vastly improves the perception of the software.

Developers and testers do tend to over-emphasise functional aspects, underlying structure and technical achievements within projects. But users are usually completely oblivious to these things – they only see what the software does, not how. Most users are only concerned about how the software will help them do their job, and how pleasant it will be to work with. They are simply not interested in the underlying structure, service layers and database optimisations.

An application may have superior architecture, awesome features and good performance, but all of these efforts will be wasted if the user interface and the user experience are inadequate. Poor design leaves users frustrated and unable to complete required tasks. A properly thought out and tested UI allows the user to concentrate on the job in hand rather than spend their time trying to work out how to use the application.

This whitepaper will list some common issues that are typically encountered when testing the UI & UX, which can be carried across multiple platforms, regardless of the user. (Of course, fundamentally the more that is known about the user the better, as this way their needs can be addressed as the product is being developed.)

Also, this paper will demonstrate to stakeholders the value of and opportunities available through investments in UX, especially when implemented at the start of the project, and how this attention to detail can be directly connected to business goals in a real and measurable way. It will demonstrate how a little attention to ‘look & feel’ drives value for a company, their product, their employees and their customers.

Good attention to UI & UX is critical to the experience of using any software application, no matter who the intended user will be. The boundaries have been pushed and the bar has been raised in the marketplace and users have become acutely aware of what is now possible with newly built applications. So really, it is becoming increasingly relevant that software should compete on quality and not just features alone. A properly designed UI, and the UX, is the differentiator that allows software to do this.

The UI, UX and usability in general are often treated as ‘poor relations’ – issues and observations often being relegated to the project backlog – but collectively, when addressed, they add high positive impact to the finished product.

2. Introduction

As functional testers, SQS are often actually right in the epicentre of UX. SQS's passion and advocacy for the user, and actions on behalf of the user, can and do influence the UX of the product. When focusing on the user rather than the feature, projects can find out what works for them, as well as what doesn't. It can be ascertained if the product is going to support their goals and help them carry out their tasks in a pleasant and efficient manner. To convince even the most sceptical stakeholders and product owners of the value of UI & UX, it is necessary to explain what UX actually is and why its contribution to a successful product is so valuable. UX is often confused with the older discipline of usability, which is chiefly about observing users working and performing tasks that the software will deal with.

Essentially, UX is the experience a user has when they use and interact with the software and how it engages them (ISO, 2010). The UI is the interface through which the users interact with the software (e.g. a web browser, a desktop application UI with windows and mouse pointers, or a touch screen device). The fundamental goal of UX is to bring applications beyond the points of user frustration, and to create user engagement. With that engagement the application can be developed according to the needs of the particular user and the aims of the functionality.

Bad UX interferes with the user's ability to focus on accomplishing their goals. Good design is actually invisible so when the product is well designed, it allows the user to focus on the task in hand, i.e. doing their job, rather than trying to work out how to use the software. Positive engagement with the application brings greater pleasure and effectiveness to the user, and negative engagement leads to frustration, disillusionment and wasted time.

UX is especially relevant in customer facing applications. To ask a stakeholder if UX should be a priority would be the same as asking them if helping and pleasing their customers is a priority. Good UX is also a good way of achieving user trust – it reiterates to them that the application is secure, provides good information, is safeguarding data, is of high quality and is the best option for them. In other words, the absence of a positive UX can lead to the conclusion that an application is insecure, does not provide valuable information and neglects safeguarding of data and so on. Consequently, the risk of users rejecting the application is high and leads to negative views on project success.

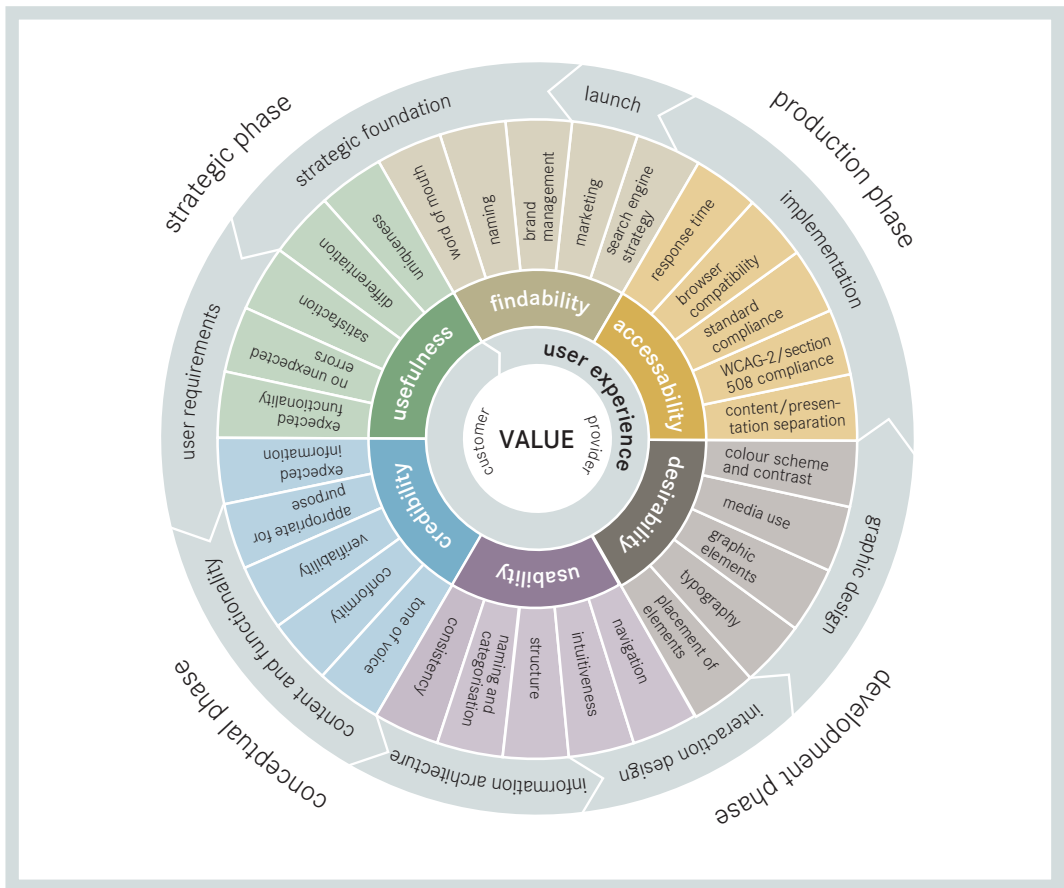


Figure 1: The 'User Experience Wheel' highlights the different methods and tools involved in UX design

According to Morville (2004), the UX is a series of different quality attributes: findability, accessibility, desirability, usability, credibility and usefulness. Revang (2009) uses a 'User Experience Wheel' (see Figure 1) to summarise the UX aspects that can be considered in projects. Business value is placed at the centre of the wheel as this is what projects have to accomplish with UX.

Clearly, for both customers and providers, positive user experience should be a project goal. As the next level of refinement shows, numerous factors have to be balanced carefully with non-UX goals and projects. The different factors are associated with different focal phases of project development emphasising that UX considerations are mandatory all across the project life cycle.

3. Market – Current Status and Outlook

Improved attention to the UI and UX can drive value for a company, their product, their employees and their customers. As part of functional testing it has become crucial in today's marketplace to raise awareness amongst stakeholders and product owners of the value and opportunities available through investment in UI and UX.

An entire generation has grown up with technology and many have become defined by their relationship to it. With older users, and also disabled users, better UX is what determines if they can use the products (i.e. hardware, software, and systems) at all.

So the pressure has never been greater on businesses to provide better UI and UX in both customer facing applications and internal information systems and consequently, the market for UI and UX will be growing over the next years. (Eckert, 2011)

That pressure isn't coming from boardrooms or stakeholders, but in fact from the users themselves. As consumers they are aware of what a UI can and should look like and as such are much less tolerant of bad UI and UX in the systems they use in the workplace. People now expect much more from software because its presence is everywhere, whether at home, work, shopping, banking, or when booking holidays. The bar has been raised which, in turn, has raised expectations for what software can do and how pleasant and engaging it can be to use.

Added to that is the fact that every day people are using the internet and being exposed to the richer UX capabilities of RIAs (Rich Internet Applications). This exposure is also having the effect of dramatically increasing user expectations for other software systems that they interact with at work and also on a personal level.

Finally, the market is also fostered by legislation – e.g. Section 508 in the US (Wikipedia contributors, 2012), BS 8878 in UK (British Standards Institution, 2010), BITV in Germany (Wikipedia authors, 2012) – that is binding for the respective public sectors.

4. Why UX is Good for Business

When a user has a positive UX it helps them perform their daily tasks with ease and generally be more happy and effective. A well designed and tested UI that is intuitive and easy to use increases productivity, reduces training costs and increases employee satisfaction. That has to be good for business.

In terms of in-house corporate software applications that are designed primarily to help employees carry out their duties, if the UX has been properly thought out it facilitates timely and accurate information as well as simplifying and automating otherwise complex tasks, which in turn increases the scope of what a single individual can manage. However, if on the same corporate application the UX has been poorly thought out, it is highly probable that the user will face issues such as

- Navigation problems
- Difficulty locating information
- Slower processing times
- Lack of feedback, help or validation input requirements
- Multiple meaningless clicks to perform a task
- Confusing menus
- Irrelevant content

all leading to reduced satisfaction and productivity. It is worthwhile pointing out that seemingly 'trivial' requirements can have an enormous impact. Example: A new interface for a call centre was rolled out and had to be used by helpdesk agents. Incoming calls were shown on the screen and the telephone IDs of the callers were displayed in a text field so that the agents could call back the clients in case of questions. The issue faced by the agents was in the colours of the displayed information – light grey

numbers on dark grey text fields. The contrast was low and the numbers were difficult to read so that the agents ran into headaches after three hours of work and could no longer operate the system.

4.1. Acquiring Support for UX

Experience shows that the hardest challenge is always trying to convince business teams to invest their limited finances from the carefully calculated budget into the UI and UX (Hurst & Grossman, 2010). These benefits can definitely be linked to the success of a business but the difficult part is communicating how our improved UX is quantifiable in terms of cost or Return on Investment (ROI).

It is therefore key to highlight the value and ROI that investing in the UI and properly planning the UX will bring for the product and their business. Focusing on UI and the UX in customer facing applications is the same as focusing on customer service and quality customer experience. A focus on UI and UX in corporate in-house systems equates to a focus on productivity, business intelligence, and job satisfaction.

Satisfying user needs and helping users meet their goals are important elements towards achieving business goals in application delivery, especially with users now expecting much more from the software systems they interact with. A generation that has grown up with technology is now largely defined by its relationship to it. For older users, people with disabilities, or less tech savvy users, better UX might be what determines whether they can use the system at all. Apple have raised the bar which in turn has raised user expectations for what software can do

and how pleasant it can be to use. Good design is ubiquitous and almost invisible and is chiefly about making an application effectively and easily serve the goals of its users e.g. 'I want to feel confident my money is stored securely'.

4.2. Example: Online Banking

It's now the norm for banks to offer online banking services for their customers. But many of these online systems were built years ago when it was a differentiating feature to actually have online access in the first place. These systems have been 'hacked' over the years to bolt-on additional features and functionality, (to keep up with competitors) – with the quality of the UX suffering as a result. So, in a case where a customer is nervous to begin with, and is then confronted with a poor UI, e.g. the design of the online banking portal is unattractive (which will portray a lack of professionalism) or it behaves strangely or performs poorly (portraying a lack of quality, trustworthiness and reliability) they will have difficulty carrying out their tasks and a lack of trust in the company as a whole will be developed.

Although the design and the UX of the portal have only an indirect connection to the security of that customer's money, they actually impact heavily on the customer's confidence and perception of the company. An online banking portal should portray a sense of security, professionalism, support and expert advice, and other qualities that would be key goals of the company's branding. Essentially the UI is a conversation between the application and the user so quantifiably this is how these aspects are measured by the user. Consequently, we have a direct connection on how UX bears on the success of a business.

Other UX links to business goals for an online banking customer might be:

- Reinforcement of branding goals
- Customer retention – higher confidence decreases the likelihood the customer will switch to another firm
- Increase in Customer Lifetime Value (CLV). A customer who has confidence with a particular company tends to place more into it over time
- Decreases in call centre and in-store customer support
- Protecting the user from accidental wrong inputs

In particular, if an application is customer facing then it is important that maximum attention is paid to the UI. In essence the UI is directly representing their brand and corporate image, so colours, logos, spacing, typography and font treatments are all elements that need to be considered. These things are so often overlooked in favour of features and functionality.

For applications that are used in-house by companies, the easier and more pleasant it is to use, and the more it helps the users do their job, the more pleasurable their working lives will be. If the application is non-intuitive and difficult to use, employees will become frustrated and disillusioned with their job. There are numerous examples of failed application rollouts due to lack of user acceptance in all industry sectors – if the employees cannot use an application (or are not willing to use it), projects may ultimately fail simply for that reason. If this is the case with a customer-facing application, the user will look elsewhere or may even write a bad review of the product online.

Accessibility is another important consideration for public facing applications, where not implementing accessibility features such as screen readers, large fonts, and contrast-colour backgrounds can alienate up to 15 % of potential users.

4.3. Business Cases for Improved UX

In most projects, investments in UI design to improve UX can be justified by one of the following obvious aspects:

- Improved productivity – If the UI streamlines and simplifies the support process, and if it improves the accuracy, timeliness and quality of the support provided to the customer, this in turn reduces labour time.
- Reduced training costs – The easier a system is to use and the more intuitively and effectively it responds to the user's needs, the less training time will be required to train staff or to on-board a new member of staff. This is especially significant in high turnover jobs such as insurance call centres and can thus decrease the costs of on-going training as well as technical and managerial assistance and mentoring.
- Improved employee retention – Properly thought out UX will reduce stress and confusion for the user, make their job more pleasant and enjoyable and thus increase job satisfaction.

In addition, it is possible to make use of all usability related attributes from ISO 25000 series (ISO, 2010) and to establish corresponding links to quantified achievements.

The return on investment for UX is greater the earlier it is implemented into the project, ideally considering and testing the UI and the UX right from the very start of the project.

Traditionally, projects bolt on the User Acceptance Testing (UAT) phase at the end of the project. If, however, a project waits until after development has finished to begin UAT, the UAT tester does not have the opportunity to positively influence the product as it is being developed. It is impractical to involve users at every iteration, but getting user feedback as frequently as is practical makes a big difference. In an ideal world, buy-in from the project should be right at the beginning. User feedback can be obtained and leverages long before having a working prototype of the final application (see Section 5.2). When we involve UAT right from the beginning of the project, it greatly reduces the risks of serious issues appearing later on in the project when they are much more expensive to correct.

If stakeholders are prepared to invest some of the budget to UX early in the project it makes such a difference to have some wireframes or screen comps. Everyone in the team (including the business) then has a unified expectation as to what will ultimately be delivered.

5. Tools for Testing the UI and UX

5.1. Getting to Know the User

When starting on a new project one of the very first things for UI and UX testing is to find out as much as possible about who will be using the application – the user. Collaborating with product owners, marketing and business teams provides an invaluable collection of information about the user’s goals, needs, insights, behaviours as well as solutions they may have to current problems and issues. If UI and UX responsible teams take the time to familiarise themselves with the user, they will be in the position to deliver a much better UX for them.

With many projects being agile today, user stories are used to communicate the needs of the user, which are essentially ‘narratives’ of selected aspects of the users’ lives and experiences that they have been bearing on their relationship to the project. The story will outline what the user is trying to achieve (the feature) and will be broken into smaller, more manageable tasks and provide a valuable source of information from a UI and UX perspective. In projects running in a more traditional mode, user stories still provide the best possible source of information and have to be derived from interviews with key users or other competent stakeholders from the project.

The stories will tell us:

- How users try to achieve their goals and perform their tasks (Lewis & Rieman, 1994)
- The issues and difficulties they encounter
- The users’ frustrations and hopes
- How the product will fit into their working lives
- A day or moment in the life of a user

and can be leveraged to streamline the UI as a starting point as well as make explicit the expectations of users towards the software or systems to be designed.

5.2. Wireframes and Screen Compositions

When trying to create awareness and a common understanding of how a feature might look to the team, written documents generally don’t do it justice. A full understanding of how proposed functionality might appear to the user can be really difficult to visualise, and usually everyone in the team will have their own interpretation. Wireframes (or rapid prototyping in the agile world) are an efficient way to test the design up-front before anything has even been built. They are invaluable for developing the team’s unified understanding of the user requirements.

Wireframes are representations of an application’s screens, workflows, and key interactions. They contain the structure and elements of an application, but the visual styling and look & feel are omitted. Leaving style out of wireframes makes it easier to focus on the functionality. So they are significantly more concrete than a written representation of an idea can be, and in being visual media they also have the additional benefit of generating a much stronger emotional reaction from the team.

The UI is about much more than making the application look ‘nice’ – it includes things like screen compositions (which are a very useful tool for testing), images and logos, fonts, type treatments, white space, colour and screen resolution. Collectively these elements support the goal of conveying

information and are things to look out for when analysing or testing the UI. On the backend, optimisation of graphics and elegant coding ensures faster processing times.

The visual design of the UI can have different tones, mood and stylistic genres depending on the product's audience and intended purpose. For example, software made for stock traders might have very subdued tones that would focus on the delivery of information without distractions from the UI design.

Following on from wireframes, Screen compositions (comps) are actual images of the screens from the application with the proposed graphical look & feel implemented. Before building the application the designers will draw these up in Adobe Illustrator or Photoshop so the key users can see exactly how the application will look before it has been built. Screen compositions are a very useful tool indeed for testing the UI, especially when they have been signed off by the business. If the comps are kept up-to-date, they can be used either in their own right or in conjunction with requirements and user stories as a test basis.

Examples for a wireframe and a corresponding screen composition can be found in Figure 2 and Figure 3 respectively.

Screen comps are invaluable as a UI test basis throughout the project to validate cosmetic consistencies and to ensure that the UI is always following branding guidelines. If descriptions are left in verbal and written media, everyone in the team will form their own perceptions and mental image about what will ultimately be delivered. This can really have a derogatory effect on the finished UI – especially if multiple developers are working on the project, each with their own interpretation of how the UI should look. Ambiguous interpretation results in subsequent cosmetic inconsistencies and in what will ultimately resemble a poorly thought out UI.

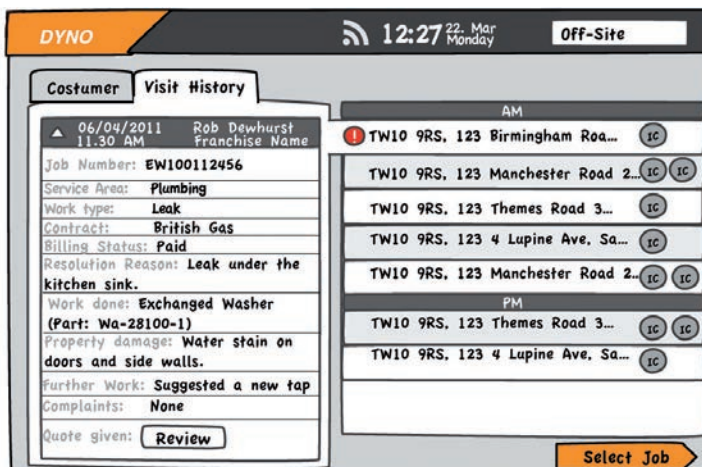


Figure 2: Example of a wireframe used for the Visit History during the Dyno Field App project in Centrica (Image courtesy of Nico Weckerle)

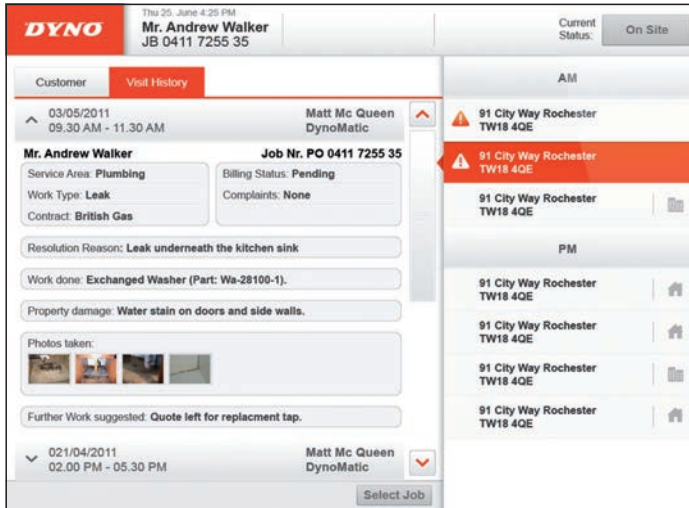


Figure 3: Following from the wireframe, this is the subsequent Screen Comp used for the Visit History during the Dyno Field App project in Centrica (Image courtesy of Nico Weckerle)

The following example shows that even light-weight implementations of the wireframes / screen comps concept are beneficial. One project focused very much on the functionality of the application with little thought and no budget allocated for UI and UX work. This was a huge concern from a QA perspective as the application was going to be customer facing. Several attempts were made to highlight those concerns to the product owner but they were fruitless. In the end, screenshots of the main screens were taken and, using tools like MS Paint and Photoshopper, were redesigned with realigned and relocated elements. After showing the product owner the updated screen comps he was then able to visualise the adjustments (rather than just imagining them) and subsequently scoped the work into the remainder of the project.

5.3. Style Guides

As well as familiarising ourselves with the user, as part of UI testing it is essential to familiarise ourselves with corporate branding guidelines (Project19, 2010). All of the large companies have style guides which, along with wireframes, comps and rapid prototyping, are a major armory in the UI tester's toolkit. They contain information about typography, RGB and hexadecimal values of colours, logo designs and scaling, spacing and kerning.

Even if you are working on a smaller project, it's worth considering creating a document that outlines the most important brand guidelines or the guidelines tied to a certain operating platform, e.g. Microsoft's guides (Microsoft, 2012) or Apple's guides (Apple, 2012) and many others (ISIT Limited Inc., 2012). This is a typical task for UI testers. By collating all the UI assets and sharing them with the team in a shared repository, it will help keep the team brand aligned. It will ensure the team is kept up to date on how colour, font and spacing should be used and will ensure cosmetic inconsistencies do not occur.

6. Testing the UI

When analysing an application's UI, paying attention to even the most seemingly trivial things to ensure the UI is polished reaps dividends. Users (and product owners) really do notice even the smallest things and what are seemingly irrelevant issues. When there is an assemblage of these issues and 'small things', collectively they convey a lack of professionalism in the product and ultimately of the team. Developers have done all their good work on the framework and the functional aspects of the application, so the functional and UI testers owe it to them, to the users and to the clients to ensure we have done our best to ensure their UI is flawless and polished. Always view the User Interface as a conversation between the application and the user!

6.1. UX Considerations when Functional Testing

The UX is something that should be considered throughout every stage of using the application, from installation, log in and first use, upgrade, maintenance and followed right through to uninstallation. If the application installation is cumbersome and contains bugs, users will assume the application itself is also cumbersome and contains bugs – before they have even used it.

An application should be responsive – there is nothing more frustrating for a user than waiting for the application to process a command – slow and unresponsive applications are not enjoyable to use – especially if progress animations are not incorporated. Users should experience perceived control as they interact with the application. As a rule of thumb, tasks that take longer than 10 seconds to

process should provide feedback of the current status to the user and also the ability to cancel. To the user, perception of speed can be very different to actual speed, and feedback from the application determines perception of speed.

6.2. Features and Functionality

When starting a new project it is useful to find out exactly what features the users need to help them achieve their goals. Business analysts and product owners can be helpful with this. They are usually very busy people and so we should always be respectful of their time. It is important to show up armed with a set of questions such as

- Discuss what tasks they do/will do
- Are there any detailed examples they could show?
- How is the application going to help them with those tasks?
- Do they have any concerns or issues surrounding the proposed application?

6.3. Screen Layout

Effective layout (i.e. the sizing, spacing and placement of content) not only makes the application more visually appealing, but is fundamental in helping users find what they are looking for in a time efficient manner. A properly designed UI can make the difference between a screen that the user immediately understands and one that leaves the user confused and frustrated.

The layout of the screen should be evenly balanced – screen elements should be evenly spaced out with an even use of white space. It’s helpful to the user if screen elements are in a place where they are expected, for example the ‘Exit’ button is usually placed at the top right hand corner of a window, and alerts are usually displayed in the centre of the screen. If there is additional information or options available, chevrons are often used to indicate this to the user without cluttering up the screen. Users only expect to see the core elements displayed on the screen by default; unnecessary objects will visually compete with more relevant ones on the screen resulting in confusion and clutter.

User controls, icons, terminology, and error messaging should be consistent throughout the interface. Where appropriate, industry and platform standards should be applied. Good designers tend to use icons with which people are familiar, rather than creating new designs which mean the same thing. Check for layout issues such as clipping of controls, text, and windows, and stretching of icons and images. Text truncation and misalignment is another common issue.

The ultimate goal when testing the User Interface is to ensure brand relevant consistency on each and every screen.

6.4. Experience Test Cases

The features should help users carry out their objectives in a clear and simple way. Ideally features will be self-explanatory and easily understood and not compete with each other. The first things to verify would be:

- **Screen design:** The screen should not be cluttered up with elements that are not relevant to that particular screen.
- **Menu design:** Menu items that are not available for the current user profile should not be shown.
- **Intuitiveness:** Key functionality should be obvious to the user.
- **Placement:** Are the right features in the right place?
- **Simplicity:** Verify that it is obvious to the user what they can do, and that only the necessary steps are displayed.
- **Ease of access:** Frequent tasks should be readily available and discoverable.
- **Readability:** The steps required to perform the task should be visible and clearly displayed.
- **Grouping:** Features that support a task should be grouped together so the task can be performed in one place.
- **Remembering user input:** With the exception of security or privacy data, it is always good to remember user input and user choices, especially when navigating backwards and forwards. However some developers do like to include a message such as ‘Your selections will be used by default in the future’.
- **Save confirmations:** If the screen allows changing of data without saving, it should prompt users to save if they move to another record or screen.
- **Delete confirmations:** If a user deletes an item, it is good practice to confirm the delete. (The exception to this would be if the UI allows deleting several records in a row, developers might consider allowing them to ignore the confirmation as it might get frustrating to keep re-clicking the confirmation message.)

- Other confirmations: Users expect to see a confirmation if an action has significant consequences and can't be easily undone – such as completing a sale or submitting a request or online form.

6.5. Error Handling

Error messages and alerts are another important consideration in application UX. They should be in layman's terms, using terminology that the user will understand. Most users won't understand technical system-worded messages such as 'Operation Failed' or 'Illegal Operation'. The words should focus on the user's goal, and not the technology. Error messages and alerts should always be polite and help the user to complete their task in an unthreatening way. Where possible, the error message should offer a solution to help the user fix the issue – names, locations and values can all be helpful. Also, error messages should not be accompanied by sound as this is unnecessarily alarming to the user. Finally, error messages should never include an 'OK' button (an error is not OK). 'Close' is more appropriate.

The following list shows words to look out for in an error message that may alarm the user, and some suggestions for a more suitable replacement:

- Illegal: Replace with 'Incorrect'
- Abort, Kill, Terminate: Replace with 'Stop'
- Error, Failure: Replace with 'Problem'
- Catastrophic, Fatal: Replace with 'Serious'
- Failed: Replace with 'Unsuccessful' or 'Unable'

There are three types of alert messages that need to be checked for consistent use:

- Error: Informs the user that a problem or error has occurred.
- Warning: Informs the user that a problem or error may occur in the future.
- Information: Provides information that is useful to the user.

Finally, error handling should be linked to logging. If fatal errors occur as users use an application, it must be ensured that the application writes those errors to a log file, event viewer, or a database table for later review and problem analysis. The application should log the routine the error was in, the person logged on, and the date / time of the error. The application should be able to handle common problems in a way that is not disruptive to the UX. Lost connection, slow connection, unavailable devices, power failure etc. should not result in applications crashing but rather an error message should be displayed that explains to the user in a non-technical way what the problem is.

6.6. UI Tests

There is a number of repeating aspects in UI testing that are valid for all projects. These are

- Cosmetic inconsistencies: The screen look, feel and design should be the same for all the screens. The Style Guide mentioned previously is a useful asset to have for this. Window frames, dialog boxes, modal windows, common controls, colours, logos, images and font families are all things to clarify. Also, capitalisation should be consistent across all screens, features and technology names. Title case is generally to be used for generic UI elements, such as the scrollbar, toolbar, menus, icons, and buttons.

- **Mandatory fields:** If a field is mandatory, it should be identified (e.g. with a red asterisk), and validation input requirements should be clearly visible to the user if the mandatory field is left blank. For example if they click 'OK' before they've entered any data the mandatory fields should be highlighted accordingly.
- **Off-brand use of colour, font and images:** Colours, logos, images and font should always be on-brand. Again, the style guide which will detail corporate fonts and hexadecimal value of colours, is the fundament for ensuring this.
- **Buttons:** Buttons should be tailored to the user – for example buttons need to be bigger on touch screen devices to avoid the 'fat finger' syndrome. The button name should not have a full stop. 'Cancel' should be used if clicking the button returns the application to its previous state; once the process is complete 'Close' or 'Stop' should be used if it leaves the current state intact.
- **Field widths:** If the screen contains text boxes that allow data entry, ensure that the width of data entered does not exceed the width of the table field. Text should never be truncated (with the exception of an eclipse, '...').
- **Data type errors:** If the screen contains dates, numbers, alphanumeric, currency or other specific data types, ensure that only the valid data type can be entered.
- **Page title:** Ideally a screen will have an identifying header to highlight its purpose, and of course it goes without saying that these should all be uniform.
- **Tab order:** If it's possible to navigate through a screen using the tab key, we would expect the tab orders to flow in reading order, which is left to right, top to bottom in Western countries.
- **Text:** Properly worded text is essential for effective communication with users. Positively worded text is easier for users to understand.
- **Spelling:** Always look for grammar or spelling errors. Spelling mistakes make such a bad impression but are so easy to avoid. Bold text should be used sparingly so the user is drawn to important text. There should not be any redundant or irrelevant text cluttering up the screen. Text should be clear and concise. Research shows that users tend to scan application text rather than deeply immersing themselves in it.
- **Text alignment:** Text alignment is another crucial and often overlooked element of the user interface. Verify that the text alignment is consistent, e.g. monetary values are right aligned, button text is centre-aligned, and so forth.
- **Abbreviation inconsistencies:** If the screens contain abbreviations (e.g. Nbr for number, Amt for amount, etc.), the abbreviations should be consistent for all screens in the application. Again, a style guide or a team wiki is helpful for ensuring this.
- **Type ahead:** If the user interface uses combo boxes (drop down lists), type ahead is a nice feature to have (if there are hundreds of items in a list, users will be able to skip to the first item that begins with that letter when they type in the first letter).
- **Scrolling:** If the application lists information in table format and that data extends past one page, the scrolling should scroll the data but leave the headers. Scrollbars should not be visible if the data is contained within one page / window.
- **Font type:** The font type should be consistent and brand specific. The font families and treatments should also be consistent, e.g. bold, italics etc. Usually the style guide determines

what this is. Bold should be used sparingly and only in areas where we want to draw the user's attention.

- Progress bars: If a screen takes more than five seconds to render results, it should contain a progress bar so that the user understands that the processing is continuing. If the user is not reassured that their request is processing, they will repeatedly click on the button they have just pressed, which often causes the application to crash. For anything that takes longer than ten seconds to process, then time-remaining estimates are useful (providing they are accurate). Progress bars should not be combined with busy pointers and should NEVER restart. If there is any text with the progress bar it should be ensured that it is displayed for a sufficient amount of time for the user to read it.
- Onscreen instructions: Any screen that is not self-explanatory to the user should contain on-screen instructions or a help option. Tool-tips are another useful way of providing more information to the user without infringing on screen real estate (just make sure they don't hover over the main text area, e.g. for vertical lists, we would place the tool-tip to the right, and for horizontal lists we would place the tool-tip at the top).

7. Conclusion and Outlook

UI and UX design and testing have become increasingly important due to users (both consumers as well as corporate users) getting more and more used to attractive UI and no longer tolerating their UX being spoiled by badly designed applications. In some areas of consumer products, UI and UX quality has become a distinctive feature – users simply drop bad designs and switch over to the better solutions. A well-designed and tested UI has tangible benefits; and in corporate environments also:

- Helps employees be happy and effective
- Helps users accomplish their goals
- Increases productivity
- Improves timelines and relevance of business data
- Increases adoption of the product
- Reduces training costs
- Reduces staff frustration and timewasting
- Increases employee satisfaction
- Helps users focus on completing their tasks instead of trying to work out how to use the software

When the UI is not designed and tested:

- The user must focus on using the software instead of achieving their goals
- Time is wasted
- Revenue is lost
- Overheads are increased
- Customer loyalty diminishes
- Brand identity is damaged

To business teams, stakeholders and product owners, a product is often merely a collection of features that give users access to a set of capabilities. But to the actual users, the product is a tool used to help them do their job or accomplish a particular goal. Although functional testers do tend to be very system focused, they are in a great position to complement testing with UI and UX aspects. In the face of technologies such as Java, Silverlight and WPF containing really quite advanced UI development libraries that enable the development of more powerful, innovative UX, the proper and solid design of UI is a must. Added to that, greater broadband availability and more powerful PCs have made it possible to solve previous performance-related UX issues. New types of devices and user input methods (iPhone, iPad, iPod, Windows 8) have given software teams more ways of creating intuitive and usable interaction designs to address new user behaviours and environments.

Historically, the most difficult and cost intensive part of building an application has been the framework, followed by features and functionality. Unfortunately, the design of the UI is often the last stage in the process, allocated whatever time and money is remaining in the budget after all the functionality has been delivered. Habitually, business requirements are centred around the delivery of features and functionality. But given the plethora of new technologies available to developers, coupled with an increased recognition that products should compete on quality and not just features alone, now is the time to embrace UI and UX as core business values and as key markers for the project that will ultimately be delivered.

8. Bibliographical References

- Apple. (2012, June). OS X Human Interface Guidelines. Retrieved from Apple Developer: <https://developer.apple.com/library/mac/#documentation/userexperience/Conceptual/AppleHIGuidelines/Intro/Intro.html>
- British Standards Institution. (2010). BS 8878:2010 - Web accessibility. Code of practice. London: British Standards Institution.
- Eckert, P. (2011, February). 2016: The User Interface Revolution Underway. Retrieved from UX Magazine: <http://uxmag.com/articles/2016-the-user-interface-revolution-underway>
- Hurst, M., & Grossmann, M. (2010, April). Taking UX Out of the Ivory Tower and Into Company Culture. Retrieved from UX Magazine: <http://uxmag.com/articles/mark-hurst-talks-about-listening>
- ISIT Limited Inc. (2012, June). User Interface Design and Usability. Retrieved from ISIT: http://www.isii.com/ui_design.html
- ISO. (2010). DIN EN/ISO 9241-210: Ergonomics of human system interaction – Part 210: Human-centred design for interactive systems. ISO.
- ISO. (2010). DIN ISO/IEC 25000 Software product Quality Requirements and Evaluation (SQuaRE).
- Lewis, C., & Rieman, J. (1994). Task-Centered User Interface Design. Retrieved from <http://hcibib.org/tcuid/>
- Microsoft. (2012, April). Guidelines. Retrieved from Microsoft Developer Network: <http://msdn.microsoft.com/en-us/library/aa511440.aspx>
- Morville, P. (2004, June). User Experience Design. Retrieved from Semantic Studios: <http://semanticstudios.com/publications/semantics/000029.php>
- Project19. (2010, June). The Importance of a Company Styleguide. Retrieved from Project19: <http://projekt19.com/the-importance-of-a-company-styleguide/>
- Revang, M. (2009, May). The User Experience Wheel. Retrieved April 2012, from User Experience Project: <http://userexperienceproject.blogspot.co.uk/>
- Wikipedia contributors. (2012, May). Barrierefreie-Informationstechnik-Verordnung. Retrieved from Wikipedia: <http://de.wikipedia.org/w/index.php?title=Barrierefreie-Informationstechnik-Verordnung&oldid=103844697>
- Wikipedia contributors. (2012, July). Section 508 Amendment to the Rehabilitation Act of 1973. Retrieved from Wikipedia: http://en.wikipedia.org/w/index.php?title=Section_508_Amendment_to_the_Rehabilitation_Act_of_1973&oldid=503498131