

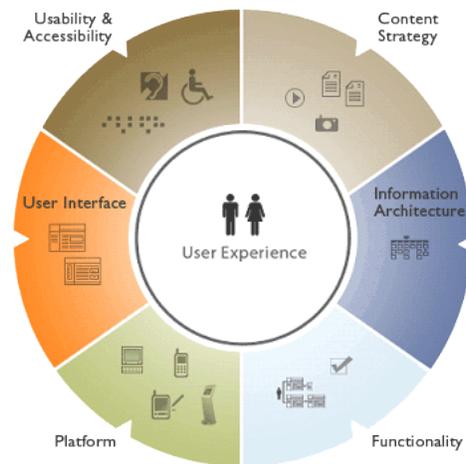
## UI/UX: Importance of Design in Software

**Introduction** – In the software industry today, there is a lot of emphasis put on design. Successful design in terms of looks and functionality together contribute to the overall success of a software. An average user of any software would undoubtedly be impressed with the features the software will provide, but at the same time the ease and intuitiveness of using the software is very important to the user as well. This article focusses on the importance of design in the user experience and interaction with a software system. We begin with a brief description of what design is and its role in software. We follow up with a high level overview of stages in a design process. We talk in detail about one of the aspects of this process – wireframes, which are heavily used in the industry today. Axure is a popular tool used for wire-framing. We then describe some techniques of design for smaller screens. This is extremely important with the growth of the mobile market.

### 1) What is Design

Design of a software system can be thought of as the process used to create the system by keeping in mind all the internals of the systems as well as the larger picture of the ecosystem of the software. It must encompass all the details pertaining to the software. The Design would possess the answers of what the system will do, how it should do it, in tandem with who will use the system. Naturally, one must take into account both **form** – this would be the looks, visual appeal, aesthetics, etc.; and **function** – this would be the utility, features, etc. of the software. The most important distinction between a Design and a Solution are that for a Design, one must ask the question of how something *might* be as opposed to how things currently are. This results in developing new innovative designs that may replace older inefficient ones or may solve a new problem very effectively.

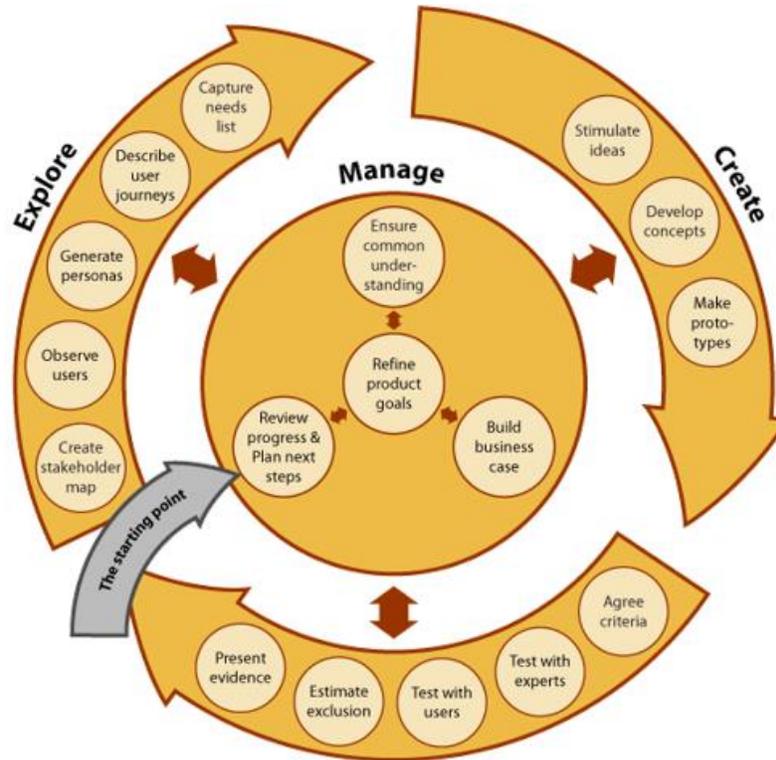
The two types of design that the industry focuses on today is User Experience design and Interactive design. User Experience design focuses on how the user would accomplish the task. This mainly deals with what the user faces. One would identify what is the best way to represent the user task at hand as an input, and also think about what the output interface will look like. Interaction design is the other side of the coin that works behind the scenes. It deals with flow of the software and what the user will experience in terms of how the software works. Needless to say, it is crucial for any software solution to have good designs for both User experience and interactions.



Datamatix makes it a point to approach both these design paradigms carefully during development. We begin by focusing on how the software would make life easier for our users. We take into account what a user would be good at doing and enable him to perform those actions thus making a software easy to use. We are always keen to hear from our clients about what they actually want and carefully identify how a quality experience could be designed to achieve those requirements. We also look forward to user testing during the design process that enables us to understand the problem at hand much better and simultaneously make the experience better. At the same time, our engineers are constantly focusing on the behavior of the software being developed and how its interaction with the user could be improved. We take into account both – what the product can **do** and how our clients would **use** them. Together, we keep track of how the user would be affected by the software and vice versa.

## 2) Interaction Design Phases

We focus on Interaction design phases in this section to understand how the behavior of a software can be extremely satisfactory. As illustrated in the figure – the three main phases would be exploration, creation and evaluation.



We begin with exploration phase. This phase mainly involves most of the user research. We look to answer questions such as who the users are, where they will be using it and how they intend to use it. The designer tries best to put himself into the user’s shoes and visualize what the user sees the system as. Some popular methods in the industry include focus groups and interviews. Apart from this, there are tools employed such as personas and storyboards.

The creation or ideation phase is all about brainstorming. The team is encouraged to come up with as many approaches as possible which would focus on depth and breadth of project scope as opposed to details of implementation. The more ideas pitched during this stage, the more it benefits the final product. It enables the team to pick the best from all proposed approaches.

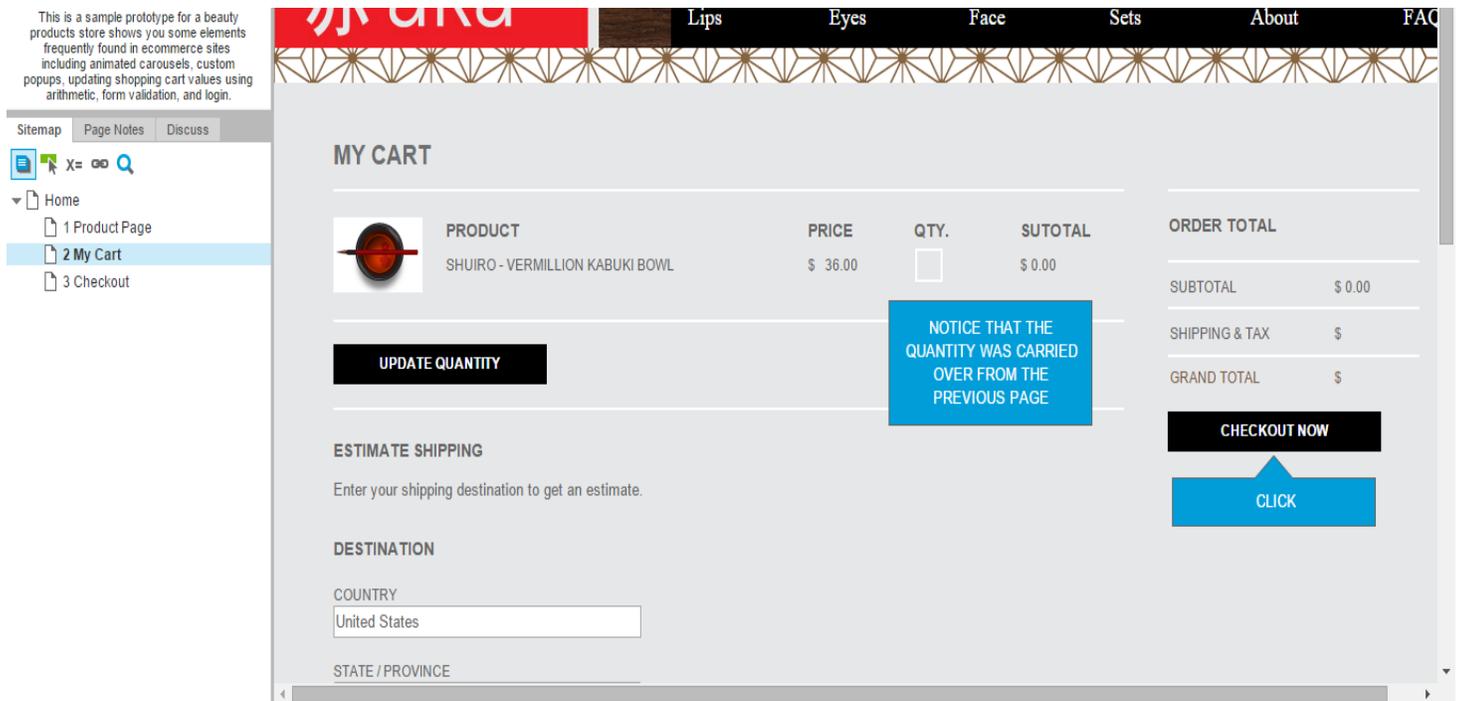
The refining or evaluation phase is the most important. Reflecting the Agile method, in this stage we must evaluate the ideas generated in the previous phase through the lens of user requirements gathered in the first stage. We must merge, sift, and build on all the combined ideas in order to best fit the above two stages. Users are asked for feedback, ideas are restructured and the process repeats until there is a consensus. This is the stage where we see prototypes being built, user studies being carried out and wireframes being generated.

This cycle for identifying the best design is very influential on the end product. It is often easy to revert early on in design than end minute alterations on a finished product. For Datamatix, these stages are crucial to understand the complex requirements of our clients and at the same time help them understand the scope of our solutions. There is no doubt that having such a model in the early stages of development vastly improves the quality of the end product.

### 3) Axure for wireframes

A wireframe is a popular tool used during the refinement phase. We will talk about them briefly here as it is very likely that a reader from the software development industry would have come across them at some point or the other. This section focuses on wireframes by describing Axure – which is a very popular wire-framing and prototyping tool.

After identifying the activities that the software must support, we must design the tasks that the user will perform in order to complete those activities. A detailed understanding of these tasks is important to understand how they will fit into the big picture together. Intuitively, one would suggest a task flow diagram that would definitely clarify the interactions between tasks. This is where wireframes come into the picture as highly sophisticated task flow diagrams. In fact, we can mockup the end product itself with a high degree of accuracy.



The image is a mockup created in Axure. The elements on this screen are all clickable and will navigate among the three pages visible in the far left of the screen. It is very easy to set up even finer details such as validations and create as close a representation of the end product. The user can then further evaluate these mockups and identify possible changes. This helps the 3 step process we described in the previous section by speeding up the process and also giving more accurate observations and results.

In this way, wireframes are actually abstracted views of the design of each screen of the system where all the planned functionalities are represented. They will enlist all the content and controls that form the structure of the system. Thus, by outlining the information hierarchy, it is a more concrete model for the user to evaluate. A single wireframe can have annotations and metadata that can give it more meaning. Linking among wireframes simulates the flow in the system page by page.

As a product development firm that is very user centric, we are always keen to deliver and exceed the expectations of our clients. By breaking complex system requirements into smaller tasks and having simpler designs we ensure that the user will receive all the functionalities he expects. Given a task within a system, we need to know how a user would start a task. Following this, how would a user normally transition to another task. Then we need to keep in mind that a user may be

lost in the system and needs to find a way back. Identify the common mistakes that may be made and recover easily from them. Also the user should understand if a task has been completed successfully. These are complex requirements that are made simple by wireframes, especially using a tool such as Axure. Datamatix employs such practices to enhance the design processes of our solutions.

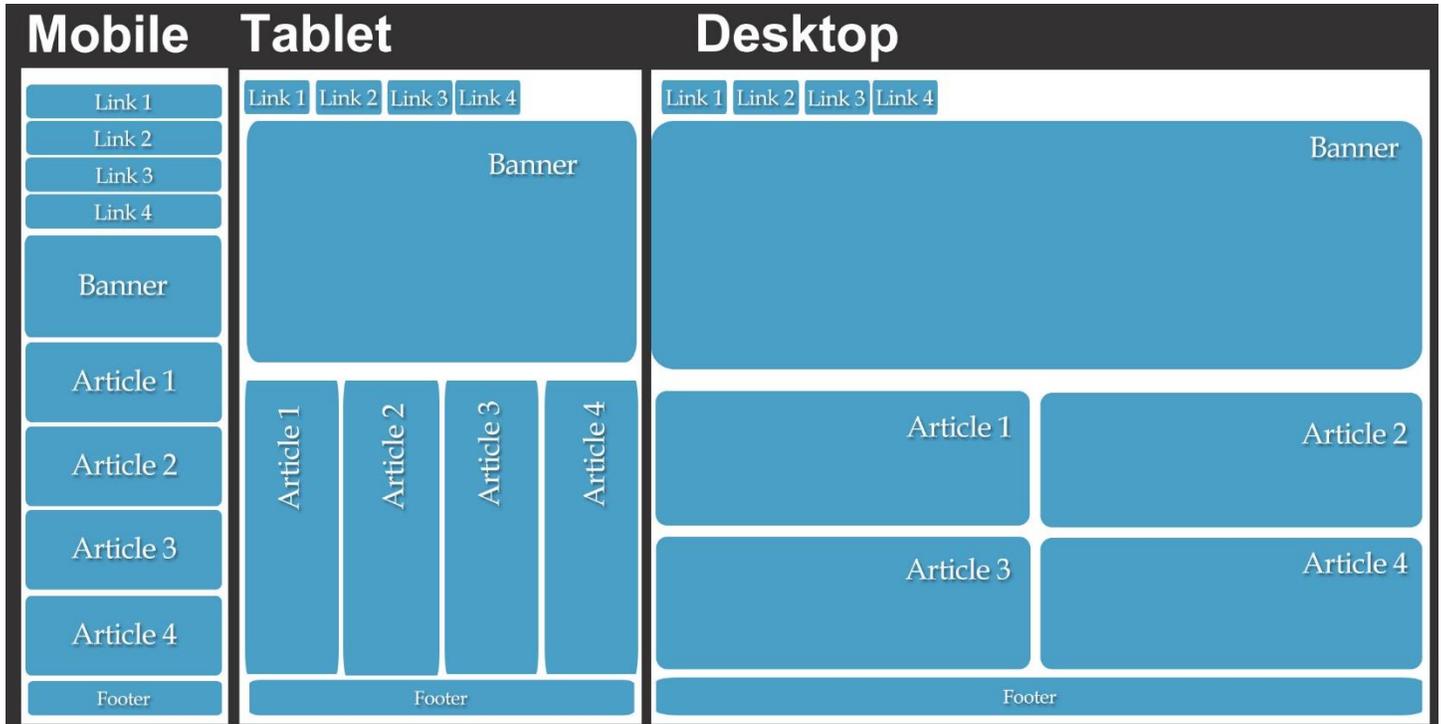
#### **4) Small screen design**

Discussions cannot be complete without a brief overview of a mobile strategy. Handheld devices will overtake desktop PCs when it comes to content search. Hence, there must be small device centric design solutions. The challenges to address for small screens is comparatively of a different nature. Screen space being smaller means a few things. Lesser text can be displayed or it is difficult to read. This translates to more scrolling. Less screen content also means more concentration from the reader to build a detailed context. Which means that there must be more focus on summarizing. After the screen size, we have the issues arising from touch. Touch means partially covering the screen. It also means widgets should be large enough so that a finger tap can accurately send the correct input.

Some of the solutions to these problems are already quite noticeable. We have a different system of tabs and pull down menus in mobile instead of pop-up windows. We see realignment of text in many websites with automatic vertical scrolling instead of a long horizontal sentence. We also see the flipping between portrait and landscape mode on a mobile where landscape is more suitable for views that require more horizontal screen size. We also have descriptive icons now instead of relying on the text below icons.

The concept that this article would like to cover is that of responsive design. The main purpose of a responsive web design is that it looks to adapt from user to user based on the device being used. There are many device dimensions in the market and a responsive design would render optimally for each device. In such a design, screen size, resolution and device type are all taken out to build an optimal viewing experience for the user. Responsive design will not only be focusing on small screens, it will respond to larger screens of tablets and desktop PCs as well.

The concepts of responsive design that we will cover in this article are media queries and flexible content. Media queries are the CSS features that fetch device properties such as device-width and pixel-density. Flexible content then adjusts accordingly based on what the media queries return. We have content grids that resize according to the screen sizes. We then have flexible lines which alter their words on a line and also the spacing between the words. Flexible images alter their quality and effectively their loading times. Navigation bars also have a very different design based on whether it is a desktop tablet or mobile screen.



**Conclusion** – Designing computer software correctly from the ground up is important for the success of a software. Design is the scaffold around which the core technical solution will be built and is often something that cannot be greatly altered once put in place. Datamatix is focused on getting these basics right in order to deliver the software of highest quality in terms of user experience and interaction along with the core technical functionalities.

**List of References –**

- 1) Axure website - <http://www.axure.com/features/>