



**T.Y. MCA
SEMESTER - V (CBCS)**

USER EXPERIENCE DESIGN

SUBJECT CODE : MCA503

Prof. Suhas Pednekar
Vice-Chancellor,
University of Mumbai

Prof. Ravindra D. Kulkarni
Pro Vice-Chancellor,
University of Mumbai

Prof. Prakash Mahanwar
Director,
IDOL, University of Mumbai

Programme Co-ordinator : **Shri Mandar Bhanushe**
Asst. Prof. cum Asst. Director in Mathematics,
IDOL, University of Mumbai, Mumbai.

Course Co-ordinator : **Mr. Shyam Mohan T**
Dept.of MCA,
IDOL, University of Mumbai, Mumbai.

Course Writers : **Mr. Dayanand Mhamane**
Assistant Professor,
G M Vedak College of Science.

: **Mr. Sandeep Kamble**
Assistant Professor,
Cosmopolitan's Valia College.

June 2022, Print - I

Published by : Director
Institute of Distance and Open Learning ,
University of Mumbai,
Vidyanagari, Mumbai - 400 098.

DTP Composed : Mumbai University Press
Printed by Vidyanagari, Santacruz (E), Mumbai,

CONTENTS

Unit No.	Title	Page No.
Module I		
1.	Introduction to UX Design-I	01
2.	Introduction to UX Design-II	18
Module II		
3.	The UX Design – Life Cycle-I	32
4.	The UX Design – Life Cycle-II	49
Module III		
5.	The UX Design Process – Understand Users	64
6.	Contextual Analysis	82
Module IV		
7.	Architecture and Interaction design of UX	95
8.	The UX design process	109
Module V		
9.	UX Evaluation and Goals	125
10.	UX Evaluation Techniques	139
Module VI		
11.	UX methods for Agile	160
12.	Agile methods from the UX Perspectives	173



T.Y. MCA
SEMESTER - V (CBCS)
USER EXPERIENCE DESIGN

Syllabus

Sr. No.	Module	Detailed Contents	Hrs
1	Introduction to UX Design	What is UX, Ubiquitous interaction, Emerging desire for usability, From usability to user experience, Emotional impact as part of the user experience, User experience needs a business case, Roots of usability.	06
2	The UX Design - life cycle	Introduction, A UX process lifecycle template, Choosing a process instance for your project, The system complexity space, Meet the user interface team, Scope of UX presence within the team, More about UX lifecycles.	06
3	The UX Design Process – Understand Users	Introduction, The system concept statement, User work activity gathering, Look for emotional aspects of work practice, Abridged contextual inquiry process, Data-driven vs. model- driven inquiry, History., Contextual Analysis, Extracting Interaction Design Requirements, Constructing Design-Information Models.	12
4	The UX Design Process	Information ,Architecture and Interaction Design and Prototyping Introduction, Design paradigms, Design thinking, Design perspectives, User personas, Ideation, Sketching, More about phenomenology, Mental Models and Conceptual Design, Wireframe, Prototyping	10
5	The UX Design Process	UX Evaluation and Improve UX Goals, Metrics and Targets, UX Evaluation Techniques.- Formative vs summative ,types of formative and informal summative evaluation methods, types of evaluation data, some data collection technics, variations in formative evaluation results, informal summative dada analysis, formative data analysis, feedback to process ,evaluation report	12
6	UX methods for Agile Development	Introduction, Basics of agile SE method , drawbacks of agile SE method from the UX perspective, A synthesized approach to integrate UX	06

INTRODUCTION TO UX DESIGN-I

Unit Structure

- 1.0 Introduction to UX Design
- 1.1 What is UX.
- 1.2 Ubiquitous interaction
- 1.3 Emerging desire for usability
- 1.4 Reference Link

1.0 INTRODUCTION TO UX DESIGN :-

User Experience (UX) is an integral part of the technology and design industry. UX aims to fulfil users' needs and provide positive experiences that keep users loyal to the product/ brand. Additionally, a meaningful user experience allows you to define customer journeys on your product that are most conducive to business success.

User experience design (UX design, UXD, UED, or XD) is the process of creating evidence-based, interaction designs between human users and products or websites. Design decisions in UX design are driven by research, data analysis, and test results rather than aesthetic preferences and opinions. Unlike user interface design, which focuses solely on the design of a computer interface, UX design encompasses all aspects of a user's perceived experience with a product or website, such as its usability, usefulness, desirability, brand perception, and overall performance. UX design is also an element of the customer experience (CX), which encompasses all aspects and stages of a customer's experience and interaction with a company.

What is UX design?

User experience (UX) is the complete journey of a user while using a platform, from login to logout. It includes all the aspects from direct interaction with the product to how it serves the purpose of completing various tasks.

Whereas UX design is a process, which teams in the industry use to create products (physical and digital) that are easy and practical to use. It is a very vast field comprising aspects of psychology, business, market research, design and technology. Every touchpoint between the customer and the company is included in creating a map for User Experience.

Difference between UX and UI Design:

In the industry, the term user interface (UI) and user experience (UX) are often used interchangeably. However, we need to understand that user interface design is not the same as user experience design.

When we talk about UI design, it refers to the actual interface of the product or the system that the user interacts with. It is concerned with all the visual and the interactive elements of the product interface ranging from the colours, typography to animations and navigational touchpoints (such as buttons and scrollbars) that the user uses.

On the other hand, UX design is everywhere and in everything that as users we come across in our day to day life. It is the experience the user derives when they interact with the product or system. It takes into consideration how the users feel, and how easy the product is for the users to accomplish their desired tasks.

The term was first coined by Don Norman. The goal of UX design is to create easy, efficient, relevant and all-around pleasant experiences for the user. As stated by Don Norman “User experience encompasses all aspects of the end-users interaction with the company, its services, and products.” UI and UX go hand-in-hand, and the design of the interface, the product has a huge impact on the overall user experience.

Factors that influence UX design

UX addresses how a user feels while using a system, whereas usability is about the user-friendliness and efficiency of the interface. UX as a field has grown and it accommodates usability. That's why it is important to pay attention to all facets of the user experience to develop and deliver successful products.

According to Peter Morville a pioneer in the UX field, there are 7 factors that describe UX:

Useful: The content/product should be original and fulfil the user's needs.

Usable: The product must be easy to use.

Desirable: Image, identity, brand, and other design elements are used to create desire, emotions and appreciation for the product.

Findable: The content/product must be easy to find onsite and offsite.

Accessible: Content/product needs to be accessible to people with disabilities.

Credible: Credibility relates to the ability of the user to trust in the product that you've provided.

Valuable: The product must deliver value to the users. Without value, any initial success of a product will eventually be undermined.

It is important for any organization to be aware of all these benefits of usability and user experience. Develop this awareness to generate productive outcomes, improve people's lives, facilitate users' interactions, and provide ease of use. The more you understand users' needs, the better your products will be. Once your requirements are clear, you can plan your next move, and design your product based on business and consumer demands.

UX designers perform a number of different tasks and therefore use a range of deliverables to communicate their design ideas and research findings to stakeholders. Regarding UX specification documents, these requirements depend on the client or the organization involved in designing a product. The four major deliverables are: a title page, an introduction to the feature, wireframes, and a version history. Depending on the type of project, the specification documents can also include flow models, cultural models, personas, user stories, scenarios and any prior user research.

The deliverables that UX designers will produce as part of their job include wireframes, prototypes, user flow diagrams, specification and tech docs, websites and applications, mockups, presentations, personas, user profiles, videos, and to a lesser degree reports. Documenting design decisions, in the form of annotated wireframes, gives the developer the necessary information they may need to successfully code the project.

Follow-up to project launch

Requires:

User testing/usability testing

A/B testing

Information architecture

Sitemaps and user flows

Additional wire framing as a result of test results and fine-tuning

A user experience designer is considered a UX practitioner, along with the following job titles: user experience researcher, information architect, interaction designer, human factors engineer, business analyst, consultant, creative director, interaction architect, and usability specialist.

Interaction designers

Interaction designers (IxD) are responsible for understanding and specifying how the product should behave. This work overlaps with the work of both visual and industrial designers in a couple of important ways. When designing physical products, interaction designers must work with industrial designers early on to specify the requirements for physical inputs and to understand the behavioral impacts of the mechanisms behind them. Interaction designers cross paths with visual designers throughout

the project. Visual designers guide the discussions of the brand and emotive aspects of the experience, Interaction designers communicate the priority of information flow, and functionality in the interface.

Technical communicators

Historically, technical and professional communication (TPC) has been as an industry that practices writing and communication. However, recently UX design has become more prominent in TPC as companies look to develop content for a wide range of audiences and experiences. It is now an expectation that technical and professional skills should be coupled with UX design. According to Verhulsdonck, Howard, and Tham, "...it is not enough to write good content. According to industry expectations, next to writing good content, it is now also crucial to design good experiences around that content." Technical communicators must now consider different platforms such as social media and apps, as well as different channels like web and mobile.

User interface designers[edit]

User interface (UI) design is the process of making interfaces in software or computerized devices with a focus on looks or style. Designers aim to create designs users will find easy to use and pleasurable. UI design typically refers to graphical user interfaces but also includes others, such as voice-controlled ones.

Visual designers

The visual designer ensures that the visual representation of the design effectively communicates the data and hints at the expected behavior of the product. At the same time, the visual designer is responsible for conveying the brand ideals in the product and for creating a positive first impression; this responsibility is shared with the industrial designer if the product involves hardware. In essence, a visual designer must aim for maximum usability combined with maximum desirability. Visual designer need not be good in artistic skills but must deliver the theme in a desirable manner.

1.1 WHAT IS UX.

The user experience (UX or UE) is how a user interacts with and experiences a product, system or service. It includes a person's perceptions of utility, ease of use, and efficiency. Improving user experience is important to most companies, designers, and creators when creating and refining products because negative user experience can diminish the use of the product and, therefore, any desired positive impacts; conversely, designing toward profitability often conflicts with ethical user experience objectives and even causes harm. User experience is subjective. However, the attributes that make up the user experience are objective.

According to Nielsen Norman Group, 'user experience' includes all the aspects of the interaction between the end-user with the company, its services, and its products.

The international standard on ergonomics of human-system interaction, ISO 9241, defines user experience as a "user's perceptions and responses that result from the use and/or anticipated use of a system, product or service". According to the ISO definition, user experience includes all the users' emotions, beliefs, preferences, perceptions, physical and psychological responses, behaviours and accomplishments that occur before, during, and after use. The ISO also lists three factors that influence user experience: the system, the user, and the context of use.

Note 3 of the standard hints that usability addresses aspects of user experience, e.g. "usability criteria can be used to assess aspects of user experience". The standard does not go further in clarifying the relation between user experience and usability. Clearly, the two are overlapping concepts, with usability including pragmatic aspects (getting a task done) and user experience focusing on users' feelings stemming both from pragmatic and hedonic aspects of the system. Many practitioners use the terms interchangeably. The term "usability" pre-dates the term "user experience". Part of the reason the terms are often used interchangeably is that, as a practical matter, a user will, at a minimum, require sufficient usability to accomplish a task while the feelings of the user may be less important, even to the user themselves. Since usability is about getting a task done, aspects of user experience like information architecture and user interface can help or hinder a user's experience. If a website has "bad" information architecture and a user has a difficult time finding what they are looking for, then a user will not have an effective, efficient, and satisfying search.

The field of user experience design is a conceptual design discipline and has its roots in human factors and ergonomics, a field that, since the late 1940s, has focused on the interaction between human users, machines, and the contextual environments to design systems that address the user's experience. With the proliferation of workplace computers in the early 1990s, user experience started to become a positive insight for designers. Donald Norman, a professor and researcher in design, usability, and cognitive science, coined the term "user experience," and brought it to a wider audience.

I invented the term because I thought human interface and usability were too narrow. I wanted to cover all aspects of the person's experience with the system including industrial design graphics, the interface, the physical interaction and the manual. Since then the term has spread widely, so much so that it is starting to lose its meaning.

—Donald Norman

There is a debate occurring in the experience design community regarding its focus, provoked in part by design scholar and practitioner, Don Norman. Norman claims that when designers describe people only as

User Experience Design customers, consumers, and users, designers risk diminishing their ability to do good design.

In addition to the ISO standard, there exist several other definitions for user experience. Some of them have been studied by Law et al.

UX is the process of creating products that provide meaningful and relevant experiences to users.

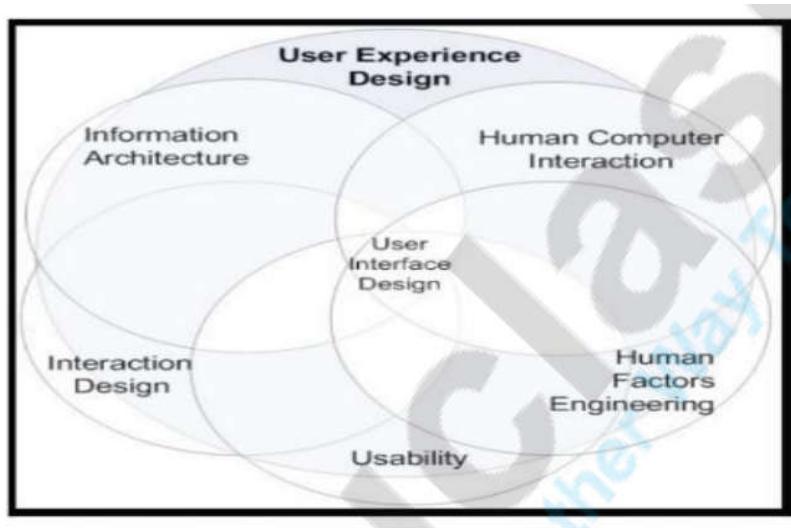
This involves the design of the entire process of acquiring and integrating the product, including aspects of branding, design, usability, and functionality.

The Aim is to design a product which we will be able to use not only with a success but with joy. To Design a product that a user will wish to use it again.

1) Look : The look of a product is all about creating a product that has visual appeal ,In other words, it has to not only look nice, but look right too. In doing so, it establishes a bond of trust and credibility between the product and the user.

2) Feel: It is really about developing products that are a joy to use. That is, whether you're interacting with them or reacting to them, products should provide a pleasurable experience and not just a functional one.

3) Usability: It is the cornerstone of user experience. If a product isn't usable, the experience of using it can never be good. And Also it cost the reputation of the brand



Research

Main article: User research

User experience design draws from design approaches like human-computer interaction and user-centered design, and includes elements

from similar disciplines like interaction design, visual design, information architecture, user research, and others.

The second part of the research is understanding the end-user and the purpose of the application. Though this might seem clear to the designer, stepping back and empathizing with the user will yield the best results.

It helps to identify and prove or disprove my assumptions, find commonalities across target audience members, and recognize their needs, goals, and mental models.

Visual design

Visual design, also commonly known as graphic design, user interface design, communication design, and visual communication, represents the aesthetics or look-and-feel of the front end of any user interface. Graphic treatment of interface elements is often perceived as the visual design. The purpose of visual design is to use visual elements like colors, images, and symbols to convey a message to its audience. Fundamentals of Gestalt psychology and visual perception give a cognitive perspective on how to create effective visual communication.

Information architecture

Main article: Information architecture

Information architecture is the art and science of structuring and organizing the information in products and services to support usability and findability.

In the context of information architecture, information is separate from both knowledge and data, and lies nebulously between them. It is information about objects. The objects can range from websites, to software applications, to images et al. It is also concerned with metadata: terms used to describe and represent content objects such as documents, people, process, and organizations. Information architecture also encompasses how the pages and navigation are structured.

Interaction design

Main article: Interaction design

It is well recognized that the component of interaction design is an essential part of user experience (UX) design, centering on the interaction between users and products. The goal of interaction design is to create a product that produces an efficient and delightful end-user experience by enabling users to achieve their objectives in the best way possible.

The current high emphasis on user-centered design and the strong focus on enhancing user experience have made interaction designers critical in conceptualizing products to match user expectations and meet the standards of the latest UI patterns and components.

User Experience Design

In the last few years, the role of interaction designer has shifted from being just focused on specifying UI components and communicating them to the engineers to a situation in which designers have more freedom to design contextual interfaces based on helping meet the user's needs. Therefore, User Experience Design evolved into a multidisciplinary design branch that involves multiple technical aspects from motion graphics design and animation to programming.

Usability

Main article: Usability

Usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Usability is attached to all tools used by humans and is extended to both digital and non-digital devices. Thus, it is a subset of user experience but not wholly contained. The section of usability that intersects with user experience design is related to humans' ability to use a system or application. Good usability is essential to positive user experience but does not alone guarantee it.

Accessibility

Main article: Accessibility

Accessibility of a system describes its ease of reach, use, and understanding. In terms of user experience design, it can also be related to the overall comprehensibility of the information and features. It helps shorten the learning curve associated with the system. Accessibility in many contexts can be related to the ease of use for people with disabilities and comes under usability. In addition, accessible design is the concept of services, products, or facilities in which designers should accommodate and consider for the needs of people with disabilities. According to the Web Content Accessibility Guidelines (WCAG), all content must follow by the four main principles of POUR: Perceivable, Operable, Understandable and Robust.

WCAG compliance

Main article: Web Content Accessibility Guidelines

Web Content Accessibility Guidelines (WCAG) 2.0 covers a wide range of recommendations for making Web content more accessible. This makes web content more usable to users in general. Making content more usable and readily accessible to all types of users enhances a user's overall user experience.

Human–computer interaction

Main article: Human–computer interaction

Human-computer interaction is concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.

Getting ready to design

After research, the designer uses the modeling of the users and their environments. User modeling or personas are composite archetypes based on behavior patterns uncovered during research. Personas provide designers a precise way of thinking and communicating about how groups of users behave, how they think, what they want to accomplish and why.[22] Once created, personas help the designer to understand the users' goals in specific contexts, which is particularly useful during ideation and for validating design concepts. Other types of models include workflow models, artifact models, and physical models.

Design

When the designer has a firm grasp on the user's needs and goals, they begin to sketch out the interaction framework (also known as wireframes). This stage defines the high-level structure of screen layouts, as well as the product's flow, behavior, and organization. There are many kinds of materials that can be involved during this iterative phase, from whiteboards to paper prototypes. As the interaction framework establishes an overall structure for product behavior, a parallel process focused on the visual and industrial designs. The visual design framework defines the experience attributes, visual language, and the visual style.

Once a solid and stable framework is established, wireframes are translated from sketched storyboards to full-resolution screens that depict the user interface at the pixel level. At this point, it is critical for the programming team to collaborate closely with the designer. Their input is necessary to create a finished design that can and will be built while remaining true to the concept.[citation needed]

Test and iterate

Usability testing is carried out by giving users various tasks to perform on the prototypes. Any issues or problems faced by the users are collected as field notes and these notes are used to make changes in the design and reiterate the testing phase. Aside from monitoring issues, questions asked by users are also noted in order to identify potential points of confusion. Usability testing is, at its core, a means to "evaluate, not create".

1.2 UBIQUITOUS INTERACTION :-

New Types of Interaction

- Our activities are supported by a wide range of devices and software.
- Our devices are capable of sensing the context of our activities.
- Our devices and services are still being designed in isolation.

Post-Desktop Model of HCI

- Information processing is integrated into everyday objects.
- Ubiquitous computing is subsuming traditional computing paradigms, for example desktop and mobile computing



Invisible Computing

- Interaction with computers should be more like interaction with the physical world.
- Users should be able to sense and control what directly interests them.
- Our devices remain the focus of attention instead of fading into the background.

People and Computers

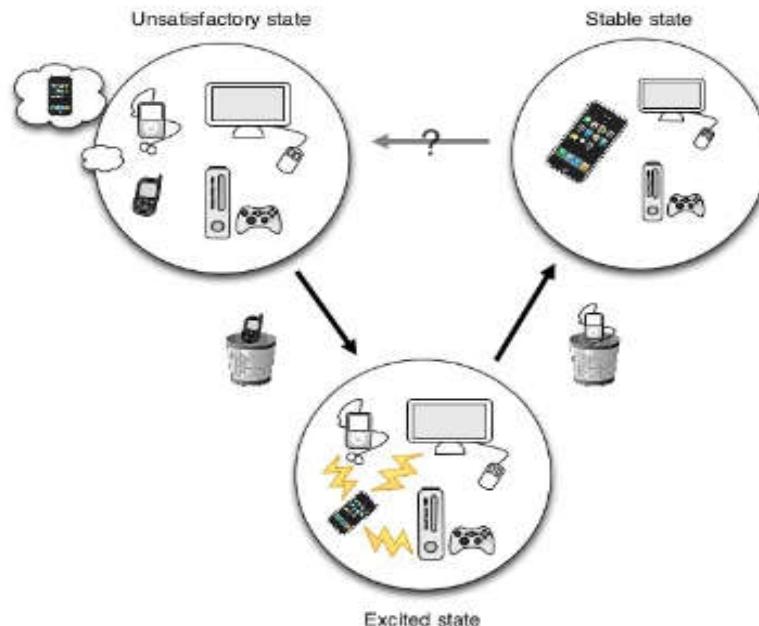
- Previously a "one-to-one" relationship, one technology - one application - one user.
- Nowadays "many-to-many", with ever-changing configurations of devices, applications, and users.

Context

- Any information, which characterizes the situation of a person, a place or an object, relevant to the interaction between a user and an application.
- Ubiquitous computing as "a technology of context".

Types of Context

- Where you are.
- Who you are with.
- What resources are nearby.



Definition

Interactions in the context of ubiquitous computing, which include multiple, dynamic, and distributed interfaces.

Properties

Occurring all the time

Occurring everywhere

- Occurring on any device with ease of substituting devices when necessary
- Being transparent
- Being informed by context

	Smartphone	Tablet	Laptop	Desktop
Communication tools	96.1%	96.9%	94.5%	94.4%
Social networking	90.9%	84.4%	87.9%	90.7%
Office applications	84.4%	90.6%	85.7%	85.2%
File sharing	71.4%	68.8%	65.9%	70.4%
Calendars	62.3%	68.8%	54.9%	61.1%
Reading	51.9%	62.5%	47.3%	48.1%
Blogging	48.1%	46.9%	42.9%	40.7%
Note-taking	48.1%	43.8%	42.9%	44.4%
Task management	40.3%	46.9%	37.4%	37%

	Smartphone	Tablet	Laptop	Desktop
Reading emails	96.1%	62.5%	96.7%	87%
Answering / composing emails	76.6%	56.3%	93.4%	79.6%
Saving draft emails	46.8%	43.8%	84.6%	72.2%
Attaching files	26%	28.1%	90.1%	74.1%
Flagging emails (as important / junk)	36.4%	37.5%	72.5%	63%
Searching through emails	62.3%	46.9%	89%	77.8%
Sending SMS messages	98.7%	15.6%	24.2%	18.5%
Receiving SMS messages	96.1%	18.8%	20.9%	11.1%
Saving draft SMS messages	76.6%	15.6%	14.3%	5.6%
Instant messaging	68.8%	43.8%	69.2%	59.3%
Voice calls	76.6%	46.9%	59.3%	40.7%
Video calls	31.2%	50%	65.9%	40.7%

The New Reality

- People use a plethora of different devices and services to support their activities.
- Devices can collect and leverage contextual information.
- We currently do not have a clear understanding how to design for this type of interaction.

The old-fashioned desktop, laptop, and network based computing systems are alive and well and seem to be everywhere with an expanding presence in our lives.

- Also complex domain systems are still the bread and butter of many business, industry, and government operations.
- Web addresses are commonplace in advertisements on television and in magazines.
- The foreseeable future is still full of tasks associated with doing computing. Although it is exciting to think about all the new computing systems and interaction styles, we will need to use processes for creating and refining basic computing applications and interaction styles for years to come. Example

Amazon Echo

Example

Amazon Echo

- Amazon Echo is a hands-free smart speaker that you control using your voice. It connects to Alexa – a cloud based voice service to play music, make calls, check weather and news, set alarms, control smart home devices, and much more.
- Echo has powerful speakers that fill the room with immersive 360° omnidirectional audio, and deliver crisp vocals and dynamic bass response.
- Just ask for a song, artist, or genre from your favourite music services like Amazon Prime Music, Saavn, and TuneIn. Using multi-room music, you can even play music across multiple Echo devices at the same time.
- With seven microphones, beam-forming technology, and noise cancellation, Echo hears you from any direction—even in noisy environments or while playing music.
- Call or message anyone hands-free who also has an Echo device or the Alexa App. Simply ask “Alexa, how do I set up calling?” to get started.
- Controls lights, plugs, and more with compatible connected devices from Philips, Syska, TP-Link and Oakter.
- Alexa is always getting smarter and adding new features and skills. Just ask Alexa to order food from Zomato, request a ride from Ola, book a carpenter from Urbancarp, and more.



1.3 EMERGING DESIRE FOR USABILITY :-

Usability is a way to measure how easy a product is to use. It is a concept in design circles to ensure products—whether websites, furniture, or hotel lobbies—can be used as simply and painlessly as possible.

For example, imagine you’re a UX designer or UX researcher building a website for a kitchenware company. Can users easily find and browse product offerings, despite never having visited the site before? Can they easily adjust items in their cart if they add something accidentally? Do customers come away from the website feeling satisfied, or confused and frustrated? These are some of the questions you’ll address by thinking about usability.

Jakob Nielsen, co-founder of leading user experience (UX) design firm Nielsen Norman Group and pioneer of usability, outlines five components that define good usability:

Learnability: A user should be able to learn to carry out simple tasks the first time they use a product.

Efficiency: Users should be able to complete tasks quickly once they’ve grasped the basic design of the product.

Memorability: Even if users don’t use a product for a period of time, they should be able to come back and remember how to use it.

Errors: A user should make few severe errors, and a product should allow users to recover from them.

Satisfaction: Using a product should be a pleasant experience.

Why does usability matter?

Good usability means users can accomplish their tasks quickly, with minimal stress and errors, and ultimately feel satisfied in their interaction with a product. For companies creating products, this becomes important for attracting customers. Users are more likely to gravitate toward products with better usability, and more likely to recommend those products to other people.

For websites in particular, usability is crucial. Visitors to a website can easily leave as soon as they encounter difficulty or confusion. When you buy a physical product, you have to go to the store or post office to return it. With a website, it’s much easier to navigate away from a less-than-ideal product.

Usability testing

Designers, user researchers, and usability specialists often run products through a process called usability testing, which can help determine what expectations, preferences, and troubles a user has. Once they have a

clearer idea of what's going well—and what isn't—they can refine a design.

Usability testing methods

It's important to conduct usability tests throughout the design process, so you can identify any potential issues as early as possible. Usability testing takes several different forms. These include:

Card sorting: Write out concepts (like features) on notecards, and ask participants to organize them into groups that make sense, then create labels for those groups. Card sorting is useful in thinking about how to organize a website or mobile app, and is often used in the mockup or wireframing stage.

Guerilla testing: A team brings a design or prototype into a public space like a cafe or park, and passersby are asked for their input. This can be a quick, low-cost way to gather feedback.

Session recordings: Often used with digital products like websites or apps, session recordings entail a researcher watching a recorded session of a user navigating the product to accomplish a task. This can also include a heatmap analysis—a visual representation of where most users are clicking, scrolling to, or pointing their mouse.

Lab usability testing: Participants are invited into a controlled environment where a moderator can observe their behavior or ask questions as they interact with a product. Since lab tests require significant coordination, and participant numbers are usually limited to small groups, lab testing is good for in-depth, qualitative research.

Remote usability testing: Participants complete a series of tasks at home. Remote usability testing can be monitored or unmonitored: If it's monitored, a user researcher is likely "watching" the participant use the product in real-time via a shared virtual space, whereas if it's unmonitored, the participant will record their session for a researcher to review later.

Determining which is the best usability test for your product depends on your budget, scheduling, and time constraints. Remote testing tends to be less expensive than in-person testing, but in-person testing can reveal a wealth of helpful information thanks to a user's body language, facial expressions, and more.

How do you measure usability?

Usability tests will ideally reveal a lot about how your product can be improved. If you're keeping an eye out for metrics on how to improve usability, consider the following metrics:

Success rate: Whether users can complete a task at all

Time: How long it takes for users to complete a task

Error rate: How many errors users made

Satisfaction: How satisfied users were

The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use", Usability came to light when it slowly turned out to be the sole reason for the failure of several software projects, both big and small. Though simple in definition, it remains ambiguous in practice since there is no way to measure the human mind and perception. Studies have shown and researches have proved that Usability is one of the key factors for a product to thrive and survive in the market. The reason for this is that Usability that was once restricted to being a measure of how efficient the User Interface of a product is, has over the years, grown beyond those shackles and now stands associated with not just the interface but the functionalities of a product as well. This new face of Usability brings with it the need to reconsider the method in which Usability was built into a project. Usability was often discussed as a non-functional requirement during the Requirements Gathering phase and then shelved until it was brought out while designing the interface for the product. But today, the whole idea of Usability has changed; it does not just focus on the final interface, rather on the entire product. Due to this, it becomes essential for developers to consider the idea of "ease-of-use" while designing the system itself. This project thus proposes a new approach of incorporating the mental model of a user, as a factor that needs to be considered while working through the Requirements and Design phases

In addition to the five components of usability, Jakob Nielsen defined the ten usability heuristics. The usability heuristics are considered rules of thumb for designers who want to create intuitive products.

The 10 heuristics are:

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recover from errors
10. Help and documentation

1.4 REFERENCE LINK :-

- [Introduction to UX design \(neebal.com\)](#)
- [UXD-qa.pdf \(mcaclash.com\)](#)
- [Usability - from desire to design | Hyrax \(calstate.edu\)](#)
- [Difference between usability and user experience | F5 Studio \(f5-studio.com\)](#)
- [From usability to user experience | Semantic Scholar](#)
- [Emotional Impact as Part of the User Experience by Wesley Hopkins \(prezi.com\)](#)
- [The Business Case for User Experience Investment | Interaction Design Foundation \(IxDF\) \(interaction-design.org\)](#)
- [The History Of Usability: From Simplicity To Complexity — Smashing Magazine](#)
- [User experience design - Wikipedia](#)
- [Usability - Wikipedia](#)
- [Ubiquitous interactions \(slideshare.net\)](#)
- [What Is Usability? Designing for Ease | Coursera](#)



INTRODUCTION TO UX DESIGN-II

Unit Structure

- 2.1 From usability to user experience
- 2.2 Emotional impact as part of the user experience
- 2.3 User experience needs a business case
- 2.4 Roots of usability
- 2.5 Reference Link

2.1 FROM USABILITY TO USER EXPERIENCE:-

User Experience

User Experience describes how end-users interact with a system such as an app or website. When one talks about user experience, it is referring to its subjective feelings and attitudes when using or interacting with a website or mobile app. It can be affected by the user interface (UI) elements, experience of interactions, brand, and emotional condition, etc. That's why, a user experience designer should analyse and anticipates the needs and expectations of the user, and then design apps and websites that best satisfy these expectations. Also, there are some UX design principles

A combination of the our experience, common practices and Value Proposition Canvas allowed F5 Studio web agency to create the effective process of UI/UX design

On the other hand, we have UX. This is a large concept that encapsulates all the “*user’s perceptions and responses that result from the use and/or anticipated use of a system, product or service*” (ISO 9241–210).

This means that not only does it take into account the technical facet, aesthetics, and brand association, but also more subjective and individual variables, like the users' expectations, previous experiences, and the context in which the product is being used.

It can also be thought of as a process. UX Design is all about projecting products that add value to our users, not only in an objective, functional way but also in an emotional and meaningful level.

It is much broader and holistic than Usability, making it more complex but also much more rewarding in the long term.

Elements of User Experience

- **System usability** – can users meet their expectations when using the system?
- **System adaptability** – do users switch to using the system to meet certain needs?
- **System desirability** – do users find the system fun, engaging and better than other systems in the niche?
- **Value of the system** – is the system valuable to its users?

Usability

Usability in Web Design and mostly in the Web development is principally concerned with the functional part of a website or app. It's the quality indicator for an interactive design. It defines how the design is efficient, easy to learn and learn with few or no errors so as to satisfy the user.

Usability is thus both objective and measurable. It measures how well products and systems satisfy their user. Usability designers can thus assess the usability of a website in various ways including surveys and feedback analysis.

UI designers should analyse a web design when determining usability, taking into account everything from accessibility and usefulness of content to credibility and designing content users will enjoy.

We can define Usability as the “*extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use*” (ISO 9241–11). It is about rethinking the design process for the users/humans (which is also the basis for Human-Centred Design):

- What do they (the users) need?
- What are their capabilities?
- What are their limitations?

Usability is a very *objective and measurable goal* because it has everything to do with performance. It is a much narrower concept than User Experience, and it is also much more logical:

- Is my product easy to use?
- Is it easy to learn?
- Is it intuitive?
- Does my product promote efficiency for the user?

- Does it prevent errors and mistakes? (which are also different by the way)

These are all characteristics of a usable product.

Elements of Usability

- **System functionality** – does the system work as it was programmed to?
- **System flexibility** – can the system perform extra tasks besides its principal function?
- **System learnability** – can users use the system without the need for extra training?
- **System design** – is the system aesthetically pleasing to its users?

Usability and user experience (UX) are considered as key quality determinants of any product, system or service intended for human use, which in turn can be considered as product, system or service success or failure indicators. At the same time, people regularly get confused between the two terms usability and user experience because they are strongly related to each other.

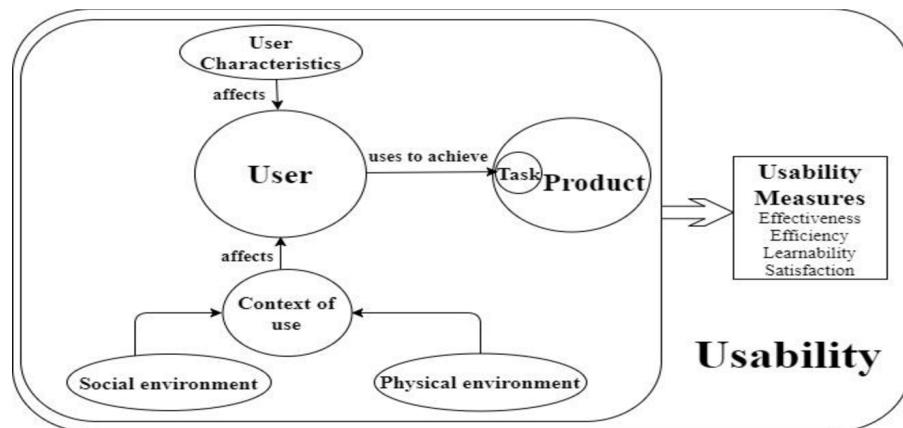
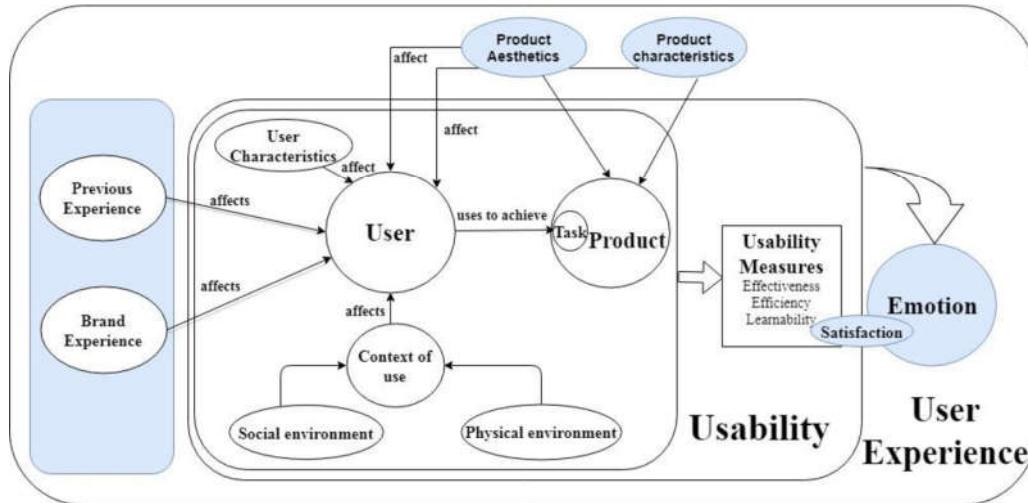


Fig. 1. Combination of Usability Definitions

2.2 EMOTIONAL IMPACT AS PART OF THE USER EXPERIENCE

Emotional aspects of user experience are the moods, feelings, and attitudes of users during their interactions with an interface. Products should not only be effective and efficient but also prompt a positive emotional response. Design is more than how something looks and functions. It has evolved to consider how you feel when you use or are about to use something.

User Experience	
Usability	
Task-related	Non- task related
Pragmatic Interaction	Hedonic Interaction
Do-goals	Be-goals
Instrumental qualities	Non-Instrumental qualities



The Potential Breadth of Emotional Impact

Not all user experiences evoke a strong emotional response. Some products may have a useful design without that something extra that sparks a deep, emotional chord of affinity. Products that really stand out to us are those with that indefinable something that transcends form, function, usability, and usefulness--something akin to the appreciation of well-crafted music or art.

IBM's design in 1998



The most successful personal electronic device ever the iPad is pointless.

It fills no obvious need for those who have a laptop and touch screen phone. Its success has little to do with rational, functional, and utility appeal and has everything to do with emotional allure. It's about the

User Experience Design personal experience of holding it in your hand and manipulating finely crafted objects on the screen.

What constitutes real emotional impact in usage?

While most emotional impact factors are about

pleasure, they can be about other kinds of feelings including:

- Love
- Hate
- Fear
- Mourning
- Reminiscing over shared memories
- Trustworthiness & credibility (important in e-commerce)
- desire for social interaction and interaction for cultural problem solving
- Compassion
- (important in sites like CaringBridge.org and CarePages.com)

Apple's design in 1998





Aesthetics and Affect

Aesthetics can induce emotions that actually increase the effectiveness of the system. Aesthetics is an elusive and confusing concept. A key issue in studies regarding aesthetics is objectivity vs. Subjectivity.

Objective View

The objective view is that aesthetic quality is innate in the object or the design and is known by certain features or characteristics regardless of how they are perceived. This means that aesthetic qualities can be evaluated analytically.

Subjective View

The subjective view of aesthetics is that it depends on how they are perceived. Aesthetics has different effects on different people and must be evaluated with respect to users/people. It's all about perceived aesthetic quality.

Combining Objective and Subjective Views to Analyse Aesthetics and Affect in User Experience

Visceral processing requires visceral design that pays attention to appearance and attractiveness. It appeals to the users “gut feeling”. Behavioural processing requires behavioural design that increases to pleasure and effectiveness . (usability and performance) Reflective processing requires reflective design that appeals to the users self-image, identity, personal satisfaction and memories.

Factors that probably contribute to emotional impact in interfaces

According to a 1998 study by Jinwoo Kim and Jae Yun Moon, these are the most significant factors in evoking emotion in users:

- Attractiveness
- Symmetry
- Sophistication
- Trustworthiness
- Awkwardness
- Elegance
- Simplicity

2.3 USER EXPERIENCE NEEDS A BUSINESS CASE

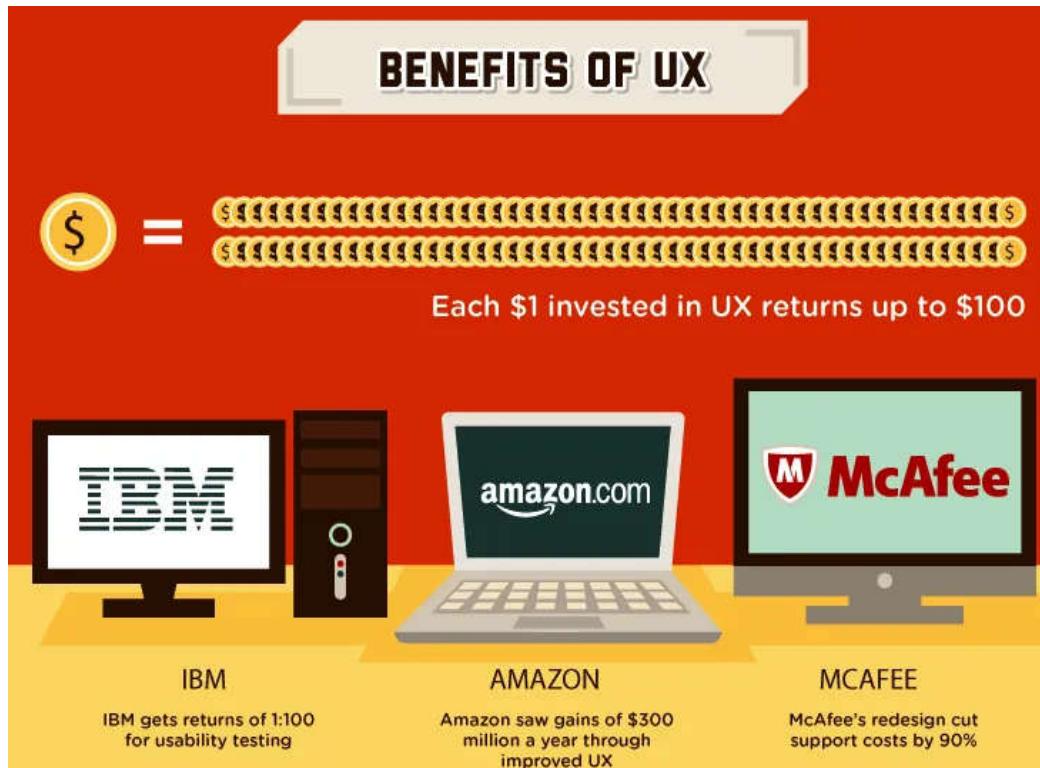
The case for user experience is clear in all our minds. Better products should lead to happier customers and more of them and thus more revenue. But how much money should be invested in user experience design as part of a software or web development project? And how do we convince those holding the purse strings that the money, they are going to spend, is going to bring value for money?

How Much Is Spent on User Experience Design?

Project investment varies a huge deal. There are apps out there which were built by someone in their spare time – the only physical cost was their time and effort and whatever portion of the life span of their hardware was expended during development. A lot of those apps get zero user experience investment and a lot of them fail accordingly.

However, when it comes to large software projects budgets can range from \$200,000 to \$20 million (or more in some cases). A rule of thumb suggests that about 40% of the total project budget should be spent on user experience research and design to produce optimal results. Now, some companies will put in a little more and some a little less but it's as good a guide as we have at the moment. We'd expect to see this figure receive a lot of revision over the coming years, though whether it will be up or down - remains to be seen.

This doesn't reflect the budgets that will be spent on marketing and selling the product once it's complete. Just the development costs of the project itself.



What Do Companies Do?

Before you can put a business case together for your 40% slice of the action; you need to think about putting your work into the context of your business. There are a small number of key activities related to the core efforts of a business. They may do other things too but these are the activities which are designed to make money from the operation of the business:

Buying things. Most companies need materials and equipment to put anything together.

Operational work. Once you have the things you need, you have to design and develop products to sell or services to offer.

Marketing and sales. Once you have your product/service it's time to tell the world about it and then extract some money for it.

Delivery. This may be simple for an internet based app business and it may be much more complex for a service delivery business. But you have to get the product in your customer's hands somehow.

After care. You won't be in business for very long if you don't supply support and care when things go wrong.

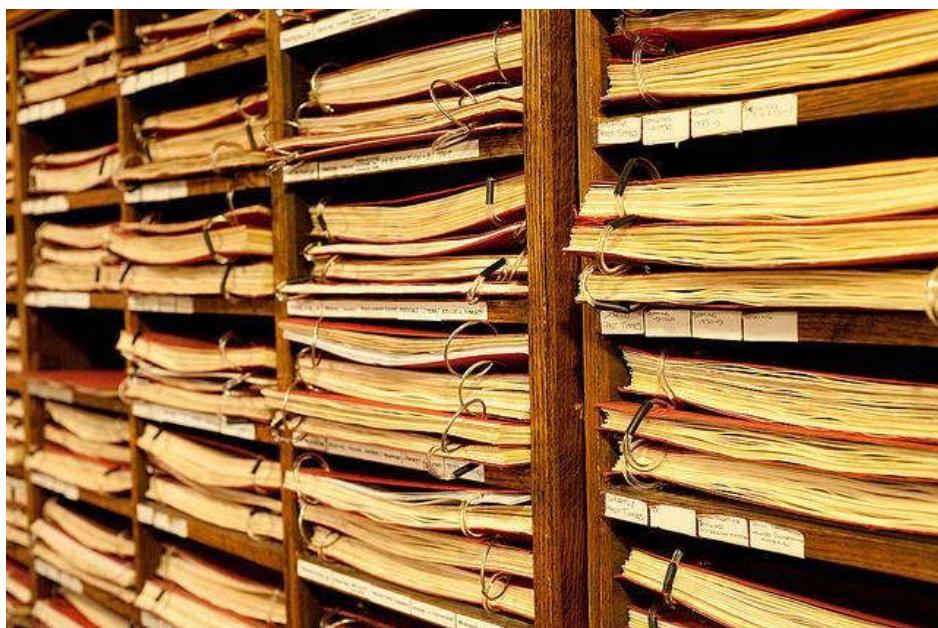
Once you understand the key drivers of business activity, you can start to link an improved user experience to them and show how you might add value to each step of the chain.

Buying Things (Purchasing and Logistics)

Not all of this category of activity involves the acquisition of physical goods. The world that most UX designers inhabit also requires a ton of information. This might be from research conducted yourself or from research carried out by an external group (such as an ethnographic research team).

The business and customer (including users) needs determine the requirements for information. UX research can provide insight into these spaces – it can help determine what contexts a product will be used in, how a product may be differentiated from another easily, it can show how your offering compares to your competitor's, etc.

In essence user research can save money in the “buying things” stage of business activity. It prevents, to some extent, mistakes being made and thus wastage. Internally focused user research can also help examine the way that the business works and how other savings may be identified in future.



Operational Work

Companies don't get anywhere without someone doing the doing. The purpose of operations, for most UX designers at least, is to design and develop the product. UX teams can add significant value to the operational work of a business.

You should be able to reduce the time taken to market by proposing multiple ideas and quickly eliminating those that don't add the value expected. You can create prototypes that allow the whole development



team to fully understand the objectives of a project reducing rework. You can ensure that the project meets user needs as well as business needs. Most importantly you can create design specifications that make it easier to build the product without people needing to ask "why" which reduces operational drag.

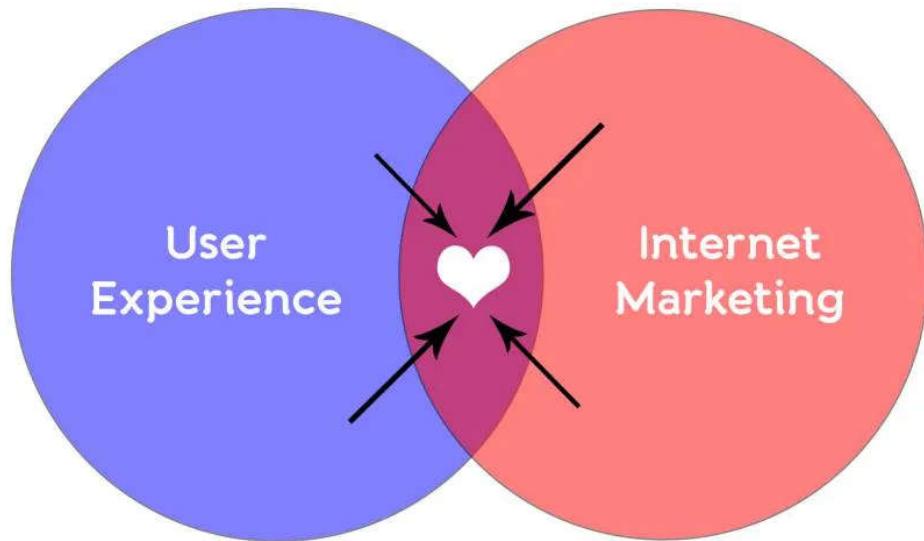
Marketing, Delivery and After Care

These three areas form a larger sphere – one of customer experience (CX) and UX is a subset of customer experience work. That means that all the work we do as user experience professionals is focused on customer experience. Not all of our users, particularly in enterprise contexts, pay the bills but all of them count when it comes to delivering a great customer experience.

It is easier to market and sell a product that was built with users in mind. If it's clear and obvious what the benefits are and that users will pay for those benefits – marketing can reduce its spending on research because much of it has already been done. It's critical that your business case shows how data will flow between the two functions.

Delivery of the product is part of the user experience too. Quick digital download versus a posted disc? It's clear which makes the process easier and less prone to error.

And it should be clear that by providing something users like and find easy to use— you'll be cutting down on calls to the helpdesk and reducing the requirement for aftercare. That's another straightforward cost saving.



2.4 ROOTS OF USABILITY :-

The concept of usability has its more recent and direct origins in the falling prices of computers in the 1980s, when for the first time it was feasible for many employees to have their own personal computer. In the 80s, most computer users had practically no, or only basic, training on operating systems and applications software. However, software design practices continued to implicitly assume knowledgeable and competent users, who would be familiar with technical vocabularies and system architectures, and also possess an aptitude for solving problems arising from computer usage.

Such implicit assumptions rapidly became unacceptable. For the average user, interactive computing became associated with constant frustrations and consequent anxieties. Computers were obviously too hard to use for most users, and often absolutely impractical. Usability thus became a key goal for the design of any interactive software that would not be used by trained technical computer specialists.

The current understanding of usability is different from the early days in the 1980s. Usability used to be a dominant concept but this changed with research increasingly focused on usage contexts. Usage quality no longer appeared to be a simple issue of how inherently usable an interactive system was, but how well it fitted its context of use.

Usability Evaluation: What's Good And What's Bad?

Usability is a contested historical term that is difficult to replace. User experience specialists have to refer to usability, since it is a strongly established concept within the IT landscape. In simplified terms, usability work is about finding out what's good and what's bad. However, when we examine the hundreds of usability evaluation methods in use, we do see that different approaches to usability result in differences over the causes

of good and poor usability. That may sound complicated but let's take two different approaches to usability:

If you think usability is a feature of an interactive system, your approach to usability may be called essentialist—i.e. poor/good usability resides in the “essence” of the system. You will typically find yourself saying things like “that website is not user-friendly”, “a website or system has poor usability when there is no visibility of system status”, “I can reliably measure which website has the best usability”, etc. This means you think that all causes of user performance are due to technology. In that case, you will typically use system-centered usability inspection methods to identify such causes.

On the other hand, if you think usability is a feature of the interaction between user, computer and the context, your approach to usability is contextual—i.e. depending on the context. This means that you think questions of user performance have different causalities, some due to technologies, and others due to some aspect(s) of usage contexts, but most due to interactions between both. Several evaluation and other methods may be needed to identify and relate a nexus of causes. You will often find yourself saying vague things like “that depends...”, “well... this website checkout procedure is great for male, fact-oriented, middle-aged buyers, but not for an impatient teenager doing the purchase on his smartphone sitting in the bus”, etc.

The reason I mention the essentialist/contextual distinction is that anyone involved with usability should—ideally—be able to say “this website/technology/system is good, that one is bad”. After all, isn't that what your client or boss is paying you to do?

To answer if the usability of a website is good or bad you have to employ a usability method. And your choice of usability method will depend on your approach to usability—whether you admit it or not. Maybe you'll deny it and say, “I've never heard of any essentialist/contextual approaches to usability.” However, that would be like selling French wine without ever having spent time in a French vineyard. You can do it, but at some point your client or boss will start asking questions you can't answer. Or your decisions will have unexpected side-effects.

So what usability method should you choose to determine “what's good and what's bad”?

The primary notion of usability is that an object designed with a generalized users' psychology and physiology in mind is, for example:

More efficient to use—takes less time to accomplish a particular task

Easier to learn—operation can be learned by observing the object

More satisfying to use

Complex computer systems find their way into everyday life, and at the same time the market is saturated with competing brands. This has made

usability more popular and widely recognized in recent years, as companies see the benefits of researching and developing their products with user-oriented methods instead of technology-oriented methods. By understanding and researching the interaction between product and user, the usability expert can also provide insight that is unattainable by traditional company-oriented market research. For example, after observing and interviewing users, the usability expert may identify needed functionality or design flaws that were not anticipated. A method called contextual inquiry does this in the naturally occurring context of the users own environment. In the user-centered design paradigm, the product is designed with its intended users in mind at all times. In the user-driven or participatory design paradigm, some of the users become actual or de facto members of the design team.

The term user friendly is often used as a synonym for usable, though it may also refer to accessibility. Usability describes the quality of user experience across websites, software, products, and environments. There is no consensus about the relation of the terms ergonomics (or human factors) and usability. Some think of usability as the software specialization of the larger topic of ergonomics. Others view these topics as tangential, with ergonomics focusing on physiological matters (e.g., turning a door handle) and usability focusing on psychological matters (e.g., recognizing that a door can be opened by turning its handle). Usability is also important in website development (web usability). According to Jakob Nielsen, "Studies of user behavior on the Web find a low tolerance for difficult designs or slow sites. People don't want to wait. And they don't want to learn how to use a home page. There's no such thing as a training class or a manual for a Web site. People have to be able to grasp the functioning of the site immediately after scanning the home page—for a few seconds at most. "therwise, most casual users simply leave the site and browse or shop elsewhere.

Usability can also include the concept of prototypicality, which is how much a particular thing conforms to the expected shared norm, for instance, in website design, users prefer sites that conform to recognised design norms.

Usability is often associated with the functionalities of the product (cf. ISO definition, below), in addition to being solely a characteristic of the user interface (cf. framework of system acceptability, also below, which separates usefulness into usability and utility). For example, in the context of mainstream consumer products, an automobile lacking a reverse gear could be considered unusable according to the former view, and lacking in utility according to the latter view. When evaluating user interfaces for usability, the definition can be as simple as "the perception of a target user of the effectiveness (fit for purpose) and efficiency (work or time required to use) of the Interface"^[citation needed]. Each component may be measured subjectively against criteria, e.g., Principles of User Interface Design, to provide a metric, often expressed as a percentage. It is important to distinguish between usability testing and usability engineering. Usability testing is the measurement of ease of use

of a product or piece of software. In contrast, usability engineering (UE) is the research and design process that ensures a product with good usability. Usability is a non-functional requirement. As with other non-functional requirements, usability cannot be directly measured but must be quantified by means of indirect measures or attributes such as, for example, the number of reported problems with ease-of-use of a system.

2.5 REFERENCE LINK :-

- [Introduction to UX design \(neebal.com\)](#)
- [UXD-qa.pdf \(mcaclash.com\)](#)
- [Usability - from desire to design | Hyrax \(calstate.edu\)](#)
- [Difference between usability and user experience | F5 Studio \(f5-studio.com\)](#)
- [From usability to user experience | Semantic Scholar](#)
- [Emotional Impact as Part of the User Experience by Wesley Hopkins \(prezi.com\)](#)
- [The Business Case for User Experience Investment | Interaction Design Foundation \(IxDF\) \(interaction-design.org\)](#)
- [The History Of Usability: From Simplicity To Complexity — Smashing Magazine](#)
- [User experience design - Wikipedia](#)
- [Usability - Wikipedia](#)
- [Ubiquitous interactions \(slideshare.net\)](#)
- [What Is Usability? Designing for Ease | Coursera](#)



THE UX DESIGN – LIFE CYCLE-I

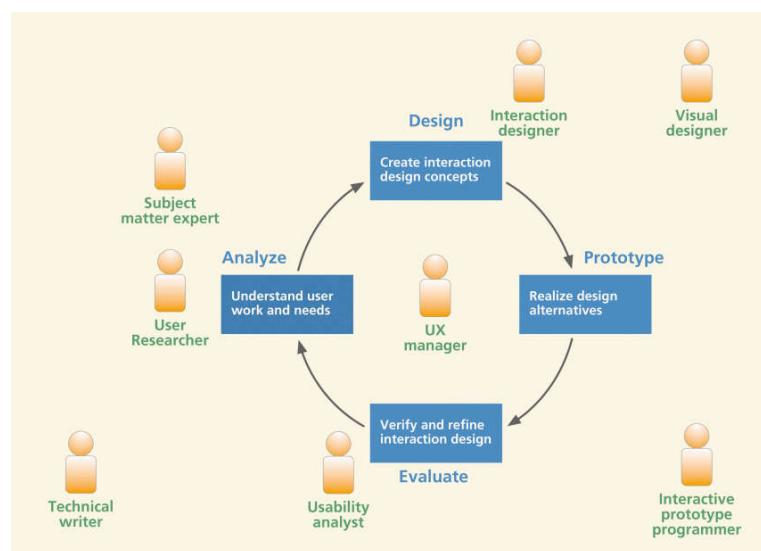
Unit Structure

- 3.1 Introduction
 - 3.2 A UX Process Lifecycle Template
 - 3.3 Choosing a Process Instance For Your Project
 - 3.4 References Link
-

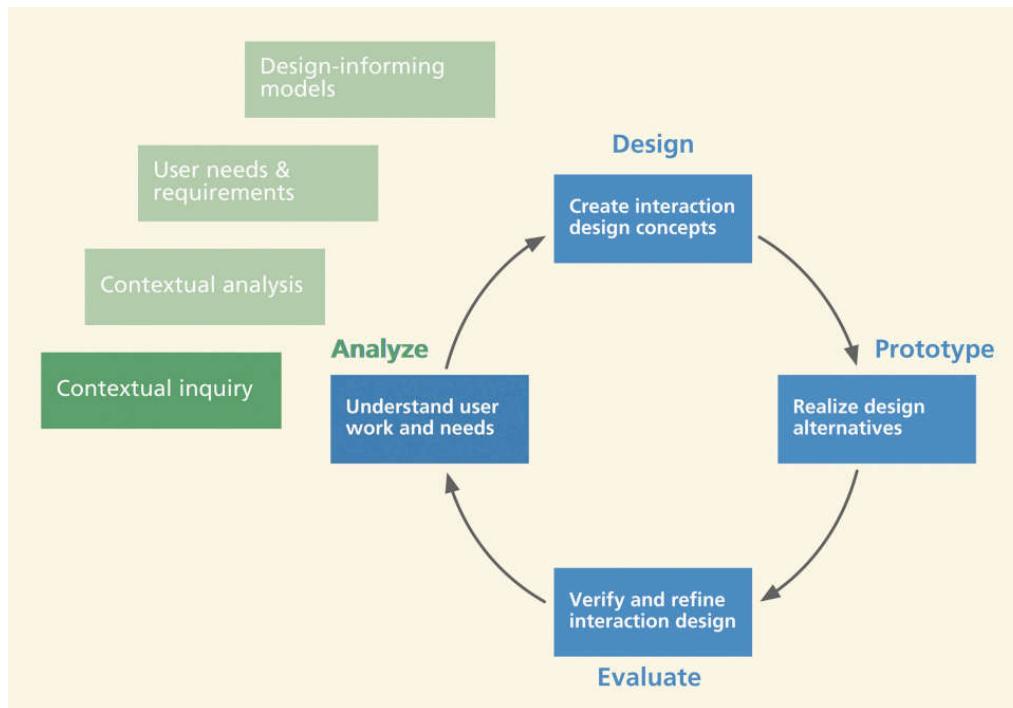
3.1 INTRODUCTION

Iterative, evaluation-centered, UX lifecycle template to evolve UX design

- Iteration: All or part repeated for purpose of exploring, fixing, or refining design
- Lifecycle is structured framework consisting of a series of stages and corresponding activities Template: a skeleton of development activities instantiated within each project Process Activities
- Analyze – Contextual inquiry – Contextual analysis – Extracting requirements – Synthesizing design-informing models
- Design – Design thinking, ideation, and sketching – Mental models and conceptual design – Design production
- Prototype – building prototypes

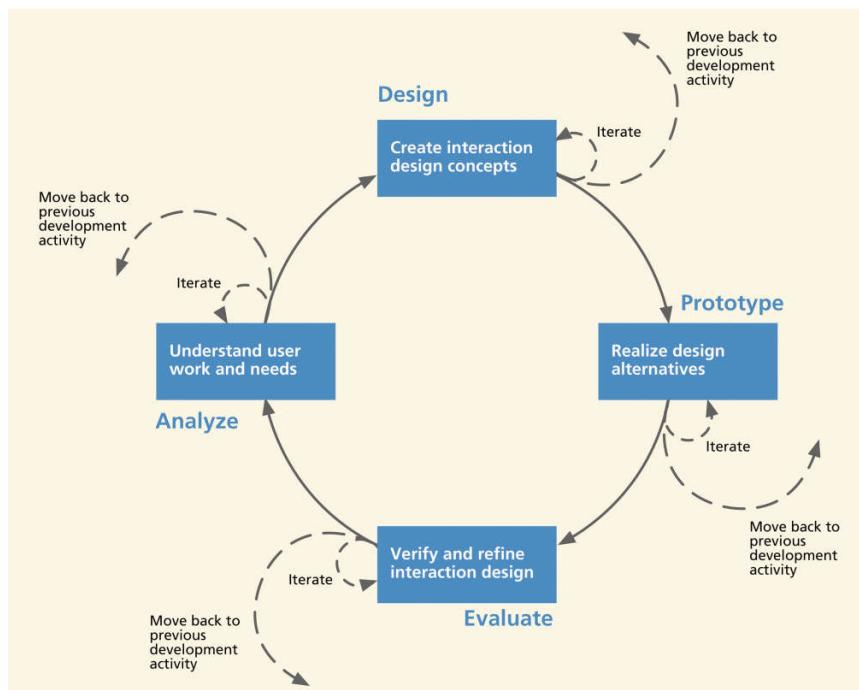


- **Evaluate – rapid and more rigorous techniques**



Contextual inquiry starts with system concept statement

- Starting point for contextual inquiry to make sure everyone is on the same page
- System (or product) concept statement – Typically 100 to 150 words in length – Mission statement for new system to be developed – Explains system to outsiders – Helps set focus and scope for system development internally



System concept statement

- Answers at least the following questions: – What is the system name?
– Who are the system users? – What will the system do? – What problem(s) will the system solve? (Be broad to include business objectives)
– What is design vision and what are the emotional impact goals?
– In other words, what experience will system provide to user?
- Especially important if target is commercial product

Introducing a running example

- Existing: The Middleburg University Ticket Transaction Service (MUTTS) – Central event ticket office, like the one on campus
- Target: The Ticket Kiosk System – Distributed self-serve kiosk system
– Especially want kiosks at bus stops around town

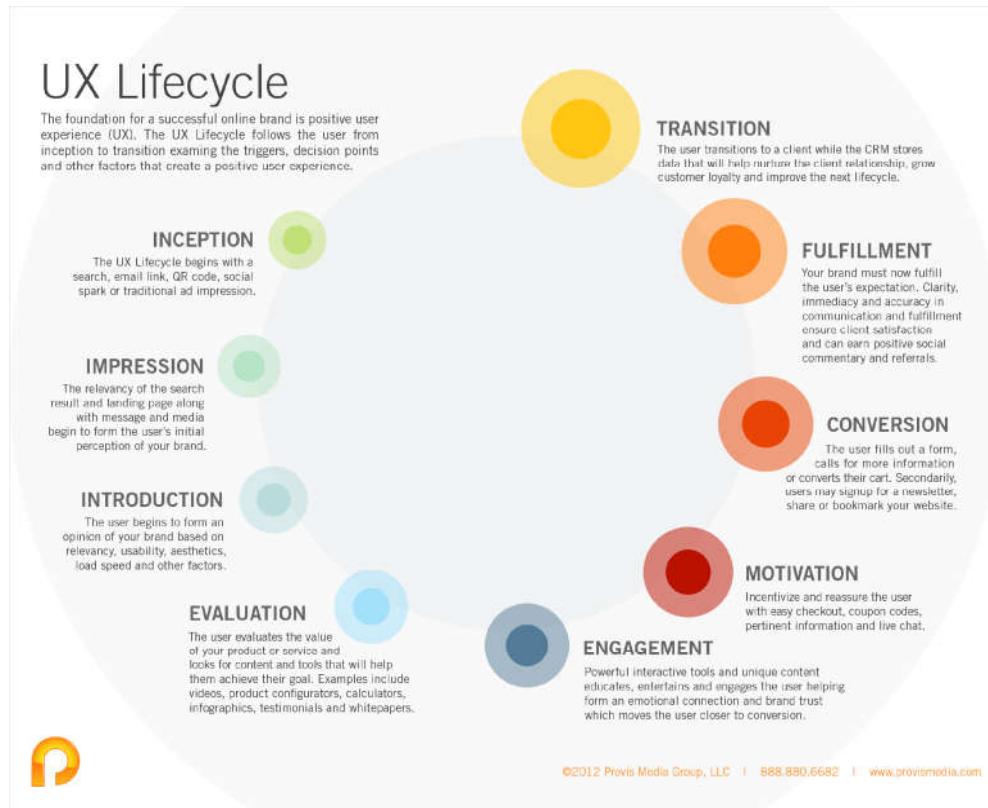
Example: For Ticket Kiosk System The Ticket Kiosk System will replace the old ticket retail system, the Middleburg University Ticket Transaction Service, by providing 24-hour-a-day distributed kiosk service to the general public. This service includes access to comprehensive event information and the capability to purchase tickets rapidly for local events such as concerts, movies, and the performing arts. The new system includes a significant expansion of scope to include ticket distribution for the entire MU athletic program. Transportation tickets will also be available, along with directions and parking information for specific venues. Compared to conventional ticket outlets, the Ticket Kiosk System will reduce waiting time and offer far more extensive information about events. A focus on innovative design will enhance the MU public profile while Fostering the spirit of being part of the MU community and offering the customer a Beaming interaction experience.

3.2 A UX PROCESS LIFECYCLE TEMPLATE

What comes to mind when thinking of your favorite restaurants? I think of a maitre d' and well dressed staff; ambient live jazz as the wine is poured into the glass; subtle lighting and tantalizing aromas and the blend of laughter with great conversation. Many would agree these aspects form a great dining experience. Of course, this experience does not happen by accident, it's the combination of staff training, culinary expertise, musical selection, lighting and everything in between. Great detail and consideration is made for each step of the experience.

Now, relate the dining experience to the user experience across your brand's digital touch points. How would potential and existing clients rate the user experience (UX) on your website, app or kiosk? To improve user experience we have to understand how it starts, where it ends and what happens between.

When building digital strategies for our clients we employ our UX Lifecycle Process which follows the user through nine steps from inception to fulfillment examining decision points, triggers and other factors that create a positive user experience. When working through this process we pose tough questions and unearth unique insights that drive innovation and improvement for your brand's user experience.



- **UX Life Cycle :-**

A positive user experience (UX) is the foundation for a successful digital brand. Our UX Lifecycle outlines nine stages of UX and identifies attributes and methods that improve perception, engagement and conversion.

- **Inception :-**

The UX Lifecycle begins with a search, email link, social spark or traditional ad impression.

- **Impression :-**

The relevancy of the search result and landing page begin to form the user's initial perception of your brand.

- **Introduction :-**

The user forms an initial opinion of your brand based on relevancy, usability, aesthetics, load speed and several other factors.

- **Evaluation :-**

The user evaluates the value of your product or service and looks for content and tools that will help them achieve their goal. Examples include videos, product configurators, ratings, calculators, infographics, testimonials and whitepapers

- **Engagement :-**

Powerful interactive tools and unique content educates, entertains and engages the user helping form an emotional connection and brand trust which moves the user closer to conversion.

- **Motivation :-**

Incentives, special offers and reassurance on value or process further motivate the user

- **Conversion :-**

The user fills out a form, calls for more information or converts their cart. Secondary conversion goals may include email signup, bookmarking and social or email sharing.

- **Fulfillment :-**

The user's expectations must now be fulfilled. Clarity, immediacy and accuracy in fulfillment ensure satisfaction, positive social commentary and set the stage for the relationship.

- **Transition :-**

Client managers and support as well as CRMs and marketing automation help nurture the relationship, and improve downstream customer experience (CX).

3.3 CHOOSING A PROCESS INSTANCE FOR YOUR PROJECT

Every systematic and consistent delivery process must be easy to learn, understand, and follow. Many designers are unsure what UX process they should use. To determine the right UX process, designers need to question their assumptions about their current process and understand the type of software development lifecycle (SDLC) their team is employing.

Design is a very reflective practice, and today's businesses demand user-centric services that focus on delivering software for local audiences that matches the user's mental model. Product development is not about implementing features and testing their usability, but about designing

products and services that are engaging and support human needs and values. Selecting the right UX process depends on many factors, including the delivery model, time to market, user experience, resource talent, team management, technology, flexibility, control, and trends.

In deciding the best UX process for your team, it is important to tailor that process to ensure it meets your needs. But you can still start with something with which you're already familiar and adapt it. Start off by asking a few basic questions:

- How stable are the requirements?
- Who are the users?
- What is the scope of the project?
- Where are the project teams located?
- Is the budget type static or operating?

Types of UX Processes

- UX for waterfall
- agile UX
- Lean UX
- pain-driven design
- UX runway

Waterfall UX

UX for waterfall is a canonical, sequential, linear method of software development in which there is no flexibility to iterate design or development. UX designers can follow a user-centered design approach, but must adhere to an agreed-on scope and timeline. Usually, there is less freedom to experiment due to this method's sequential flow. The budget for a project is estimated and usually predetermined or limited, so the waterfall-UX approach could encounter a lot of challenges in execution, especially if there are deviations from the process. During research, it is best to glean all user stories and get necessary stakeholders involved right from the project's inception.

When Should You Use Waterfall UX?

Use waterfall UX in the following cases:

- You're working on a short-term, fixed-bid project.
- The team has good experience executing similar projects.
- The requirements are stable and fixed.
- The team is experienced with the technology.

- The scope of the project is large.
- You have a fair idea of what you need to do.

Challenges for Designers

Waterfall UX presents the following challenges to designers:

- longer reviews, signoffs, and rework times
- detailed deliverables followed by documentation
- visual designs with detailed style guides for development
- wireframe flows detailing every interaction and styles
- usability testing that might have no immediate impact on the design
- involving or coordinating with usability specialists early during the requirements-gathering phase
- ensuring confirmation of acceptance by stakeholders

Any dichotomy between time and value depends on the execution plan. Because of waterfall's linear approach, a UX team working in this environment is often subject to unexpected requirements and technology constraints that impact the feasibility of design solutions. Clients may worry that the product could change direction once they receive the results from UX research. Plus, budgets tend to increase toward the end of the SDLC if the team spends too little time on research at the project's inception. Figure 1 shows the sequential design and development phases of waterfall.

Figure 1—Aligning UX with the waterfall model

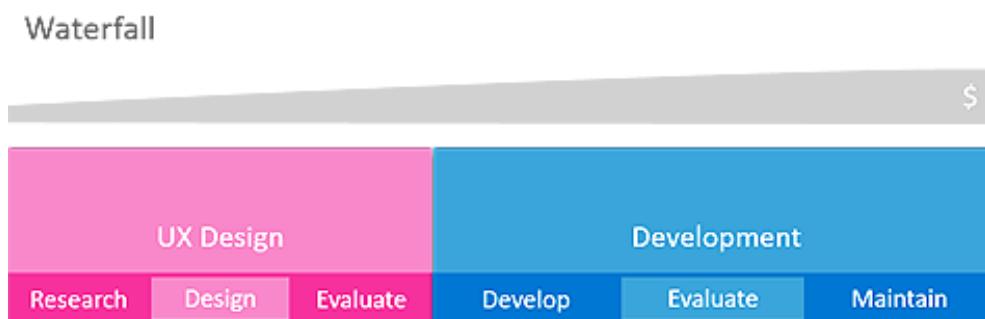


Figure-1

Modified Waterfall Models

There are many versions of modified waterfall models. One of the most popular variations is shown in Figure 2. In this version, teams have the flexibility to iterate previous phases and dig more deeply during certain phases. This flexibility can enable a team to redefine some of their more speculative solutions.

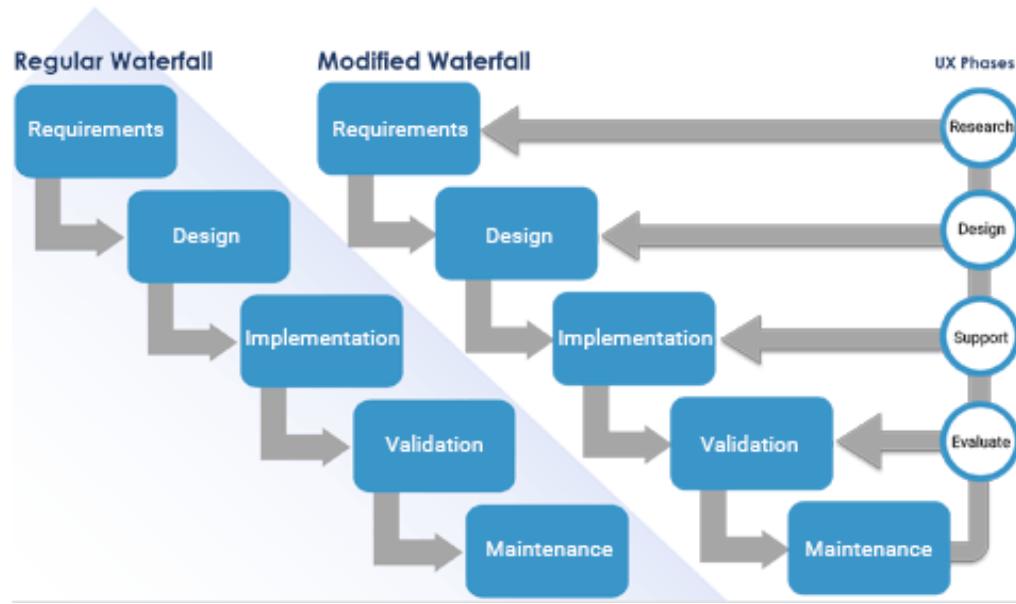


Figure-2

Agile UX

In recent years, agile has become the most popular software development methodology in the industry. It is best for scenarios in which requirements change frequently and teams are doing iterative development. Agile focuses on development and timelines. It also embodies a periodic, incremental development approach to solving problems.

A central team of designers, researchers, and business analysts work on the initial discovery stages, then hand over the requirements to the UX designers and Scrum team, who implement a design solution. Agile UX focuses on building a product backlog comprising the following:

- product vision
- experience diagrams
- journey maps
- business and UX strategies
- market research
- functional-requirements document
- competitive analyses
- stakeholder maps
- usability-testing analysis for any legacy systems
- ecosystem maps
- feature list
- feature-prioritization matrix
- personas

User Experience Design Design works ahead of development by a minimum of one sprint. During sprint planning, the team prioritizes the list of features; then a Scrum team selects product-backlog items, moves them to the sprint backlog, and depending on the velocity of the team, implements them. UX designers plan and facilitate collaborative activities such as design studios and brainstorming sessions with a Scrum team, which comprises a Scrum master, developers, product owners / managers, and technical writers.

At the end of each sprint, the Scrum team demos its outcome during a sprint-review meeting. This helps the team to assess how well their overall progress aligns with the product roadmap.

When Should You Use Agile UX?

Use agile UX in the following cases:

- You need to test against fixed outcomes.
- You must ship products quickly, based on a timeline.
- You require the ability to scope, descope, or prioritize features based on budget, quality, and value through a number of sprints.
- You need to act on customer demands over time.
- You must ensure user-friendly products by collaborating with customers during development.
- Your intent is to achieve customer satisfaction through rapid, continuous delivery of minimum viable experiences (MVEs).

Challenges for Designers

Agile UX presents the following challenges to designers:

- UX designers and project teams having a proper understanding of their role within an agile team
- aligning the entire team to communicate a common vision for the product
- tunnel vision due to the team's pace. Sometimes teams are so focused on executing user stories that they don't look at things from a broader perspective.
- a development-focused approach
- UX designers having too much to do within a two-week sprint
- absence of UX research before the start of sprint 0
- no proper induction or training for designers at the beginning of a project
- staying ahead of the sprints
- too many responsibilities

The success of a project's outcome is dependent on validating designs by receiving continuous feedback from users and stakeholders either at every sprint or for key releases. When the scope of the product is large, there might be multiple Scrum teams on the project. In such cases, a central UX

team governs the overall process. This UX team should also be involved during the initial discovery and in evaluation sessions throughout the SDLC.

The UX Design – Life Cycle-I

Visualizations of the Agile UX Process

Figure 3 shows how an agile UX process works. (Click the image to view an enlarged version of Figure 3.) I've modified this visualization of an agile UX process from a blog post by Krish Mandal of Tallan. Figure 4 shows another visualization of the agile UX process by Jeff Gothelf.

Figure 3—A visualization of the agile UX process

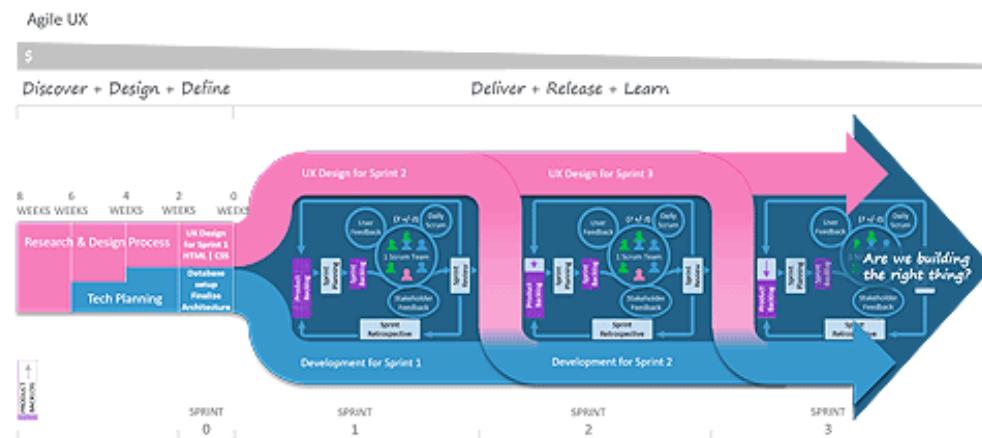


Figure-3

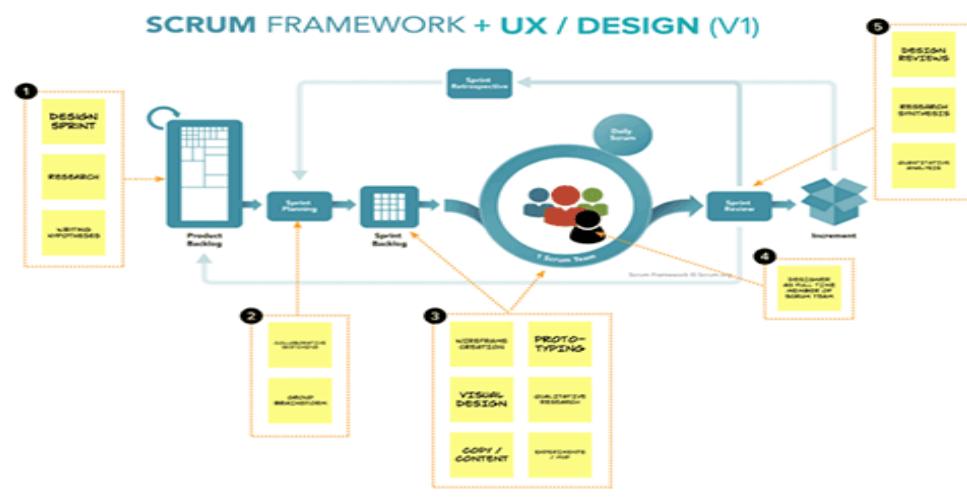


Figure-4

Organizations can also consider using SAFe (Scaled Agile Framework) in the following cases:

- When a team wants to replicate the success of their current agile process across multiple programs and portfolios.

- When there is a need for teams to work independently, to scale agile across the organization, roles, portfolio, and programs.
- When a team experiences gaps, delays, and failures with their current agile process.
- When it is necessary to improve development lead times.

Lean UX

Lean UX derives from Lean manufacturing, which focuses on eliminating wasteful production of unwanted features. Its build, measure, learn loop lets teams iterate their designs based on user feedback—across multiple sprints—to build a minimum viable product (MVP). Lean UX follows an iterative, participatory design process that ensures equal participation by design and development team members and generates many ideas and possible solutions to problems. Since, as shown in Figure 5, the entire team is involved throughout the entire process, there is a lot of team learning, which prevents the emergence of silos and gurus. A set of 12 principles drives Lean UX, which focuses on co-creation and quickly iterating designs, then validating these experiments during each sprint. There are a few challenges in adopting Lean UX—for example:

- When there is no requirements proliferation, consensus mechanisms can sometimes force compromise or cause team members to misjudge designs and eliminate their most innovative elements.
- Lean UX may work best for collocated teams rather than distributed teams. However, you can solve this problem by ensuring the adoption of better research and design collaboration tools and approaches. A few of these tools include the following:
 - UserZoom
 - Usability Hub
 - User Testing
 - Optimal Workshop
 - Dscout (Mobile UT)
 - Ethnio
 - Inspectlet
 - Focus Vision
 - Optimizely
 - Look Back
 - Usabilla
 - Realtimeboards
- For large-scale projects with multiple modules, shorter sprint cycles and a quick-fail approach may frustrate team members.

Figure 5—Teamwork is inherent in the Lean UX process

The UX Design – Life Cycle-I

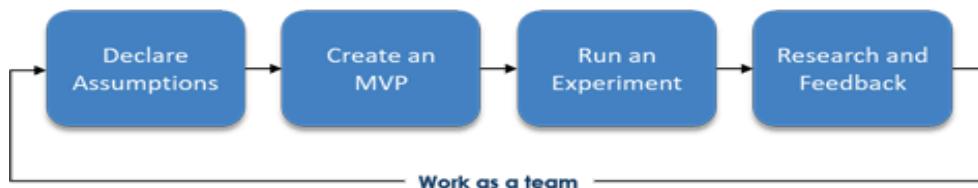


Figure-5

When Should You Use Lean UX?

While there are many situations in which you could use Lean UX, I've limited the following list to a few key points. Use Lean UX in the following cases:

- When there is a need to build products at startup speed and agility is prized.
- When a team needs to conduct experiments—by iteratively creating MVPs—whose outcomes result in usable products.
- If a team wants to get out of the deliverables business.
- If a team wants to produce less documentation.
- When choosing the right approach to product success requires a large learning curve.
- When most team members are collocated.
- When an organization is already following an agile process.

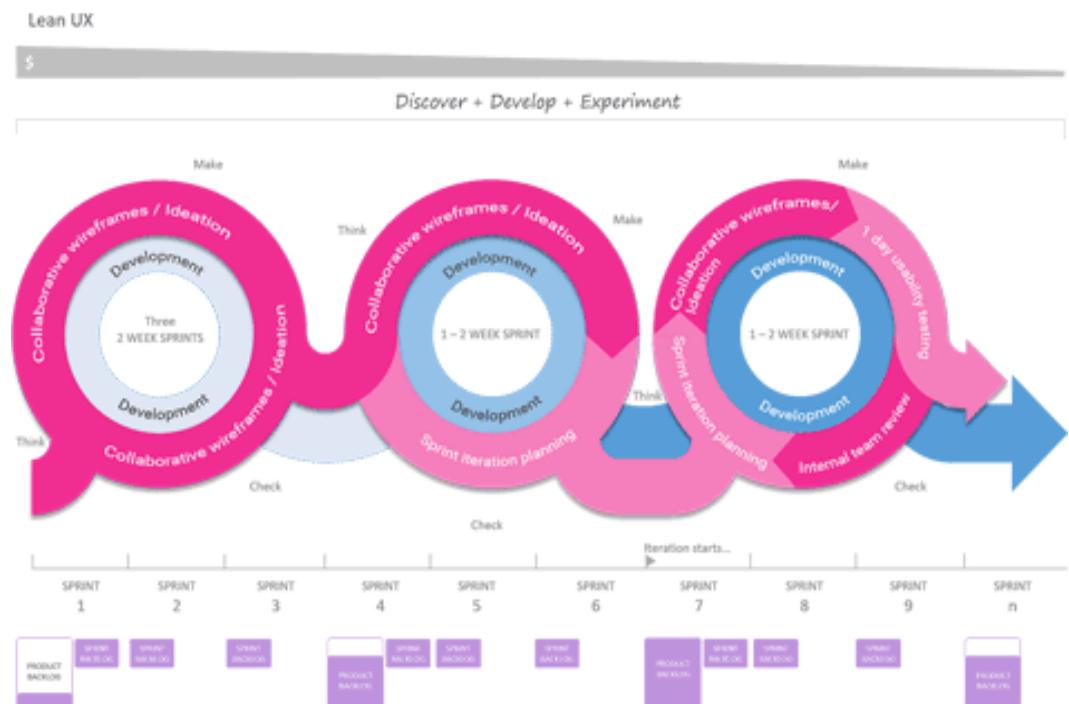
Challenges for Designers

Lean UX presents the following challenges to designers:

- continuously staying focused during iterations
- creating and tracking multiple prototypes in parallel
- defining project boundaries with the product owner (PO)
- negotiating releases with the product manager
- understanding and constantly communicating the vision for the product
- choosing the right facilitation and communication techniques for the audience—especially when the team is not collocated
- building a custom framework by modifying a process that works for the team
- having some team members distributed across geographies

Figure 6 shows how Lean UX works in tandem with iterative design sprints. (Click the image to view an enlarged version of Figure 6.)

Figure 6—Lean UX process



Pain-Driven Design

Pain-Driven Design (PDD) is all about figuring out what's causing pain before you start solving design problems—that is, predicting pain, then responding to it. PDD enables organizations' acquisition of knowledge during the discovery phase—and lets startups fail safely. Most successful, mature organizations already follow this approach either directly or indirectly—without knowing it's a pain-driven design process. Engineering-dominated organizations obtain feedback from their customers, then implement their customers' requests. Often, stakeholders become biased by customers' requests or Support or Sales teams' feedback from the marketplace.

Directives of PDD include the following:

- early validation
- seeking external collaborations within the ecosystem
- continuous learning
- disaster prediction
- pain-driven design and development

PDD follows the Lean Startup approach of validating the product early to ensure its success. Both PDD and Lean UX recommend building a hypothesis and validating it with users. However, the key difference between them is the point at which validation occurs. Lean UX teams test their hypothesis through experiments during the course of a sprint cycle.

PDD focuses on prediscovery, as Laura Klein suggests in her book *UX for Lean Startups*. There are many successful examples of teams employing PDD, but the most interesting cases are buffer and Dropbox.

Joel Gascoigne, the founder of buffer, validated his product idea by launching three screens, shown in Figure 7, to understand whether someone would be interested in paying for the idea. He had initially committed to one week and eventually launched the product in seven weeks.

The figure consists of three vertically stacked screenshots of the Buffer website, each with a large green arrow pointing downwards from the top one to the bottom one, indicating a progression or validation cycle.

- Screenshot 1:** The title "Tweet more consistently with **buffer**" is at the top. Below it is a numbered list:
 - 1** Choose times to tweet.
For example, 3 times a day at 9:30, 13:30 and 17:30.
 - 2** Add tweets to your buffer.
Manually or with our handy browser extensions.
 - 3** buffer does the rest. Relax.
We tweet for you. Just keep that buffer topped up!
 A green button labeled "Plans and Pricing" is located in the upper right corner. A small copyright notice "© 2010 buffer. All rights reserved." is at the bottom left.
- Screenshot 2:** The title "Tweet more consistently with **buffer**" is at the top. Below it are three plan options:

Free	Standard	Max
\$0/mo	\$5/mo	\$20/mo
Who? Dip your toes. Find your feet.	Who? Great for most users.	Who? Best for business & enterprise.
1 tweet per day	10 tweets per day	Unlimited tweets per day
5 tweets in your buffer	50 tweets in your buffer	Unlimited tweets in buffer

 A green button labeled "Plans and Pricing" is located in the upper right corner. A small copyright notice "© 2010 buffer. All rights reserved." is at the bottom left.
- Screenshot 3:** The title "Tweet more consistently with **buffer**" is at the top. Below it is a message: "Hello! You caught us before we're ready." followed by a paragraph about Buffer's progress and a form to enter an email for a reminder. A green button labeled "Remind me" is part of the form. A small copyright notice "© 2010 buffer. All rights reserved." is at the bottom left.

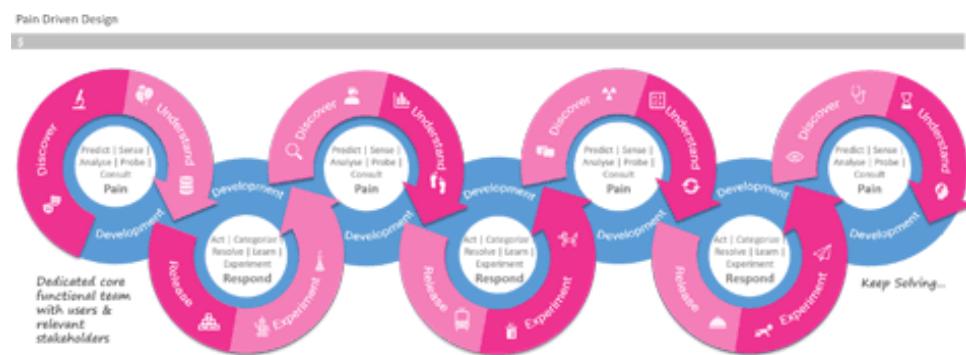
A recommended approach to PDD is to focus on problems that are associated with users' functional jobs rather than on features and functionalities. As Harvard Business School Professor Theodore Levitt has said: "People don't want to buy a quarter-inch drill. They want a quarter-inch hole."

Integrating PDD into your development process—by breaking down the complexity of pain-driven decision-making into actions lets you articulate the differences between predictable problems and unpredictable pain. Actions for quick turnarounds and building hypotheses include the following:

- Understand
- Consult
- Respond
- Predict
- Sense
- Experiment
- Act
- Categorize
- Analyze
- Learn
- Probe
- Design

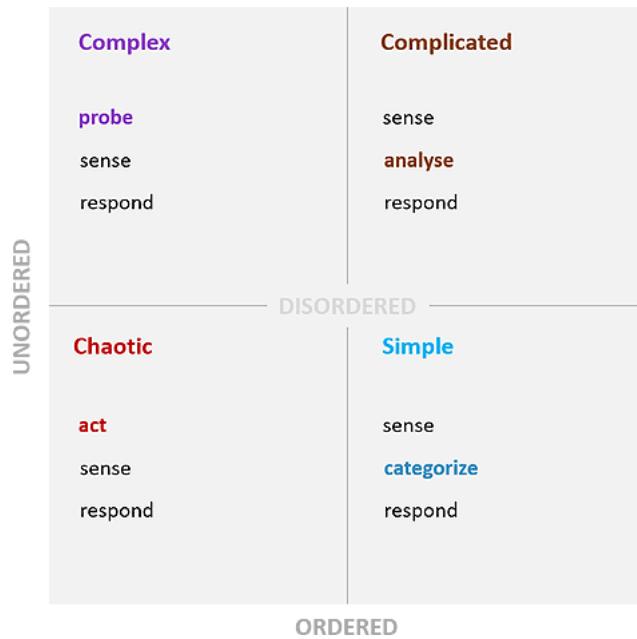
You'll arrive at a process that is similar to Lean UX, but with the flexibility to modify your actions depending on your decisions. Based on my experience, I have tried to depict this approach as the process shown in Figure 8. (Click the image to view an enlarged version of Figure 8.) The result is a dedicated team whose focus is on resolving painpoints for the organization's customers. This team consists of Champions from various departments within the portfolio, including User Experience, Technology, Sales, Marketing, Finance, and Management. The Chief Experience Officer (CXO) leads this team.

Figure 8—Modified pain-driven design model



Also, consider the Cynefin framework shown in Figure 9.

Figure 9—Cynefin framework

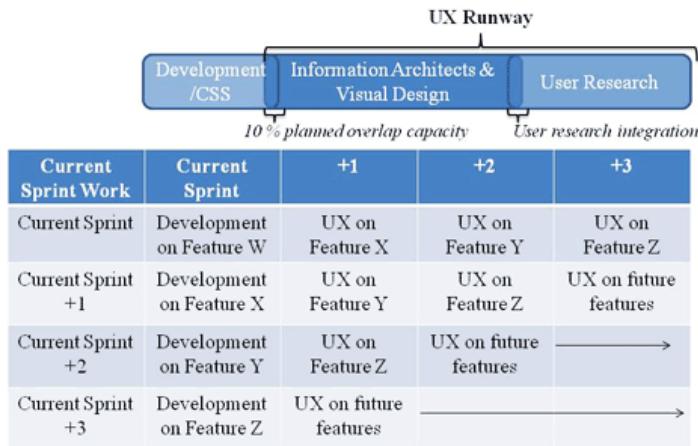


UX Runway

The idea for UX Runway originally came from the [Scaled Agile Framework \(SAFe\)](#). Many people complain that agile works well only for small-scale projects and doesn't scale up well. SAFe can provide a solution to this problem because it adopts both Lean and Agile principles, combining them to work effectively for large-scale projects that involve multiple Scrum teams. SAFe processes let you scale efforts across teams and leverage processes at the program and portfolio level.

UX runway works best with SAFe and focuses on keeping UX lean and agile at both the team and program level. Its focus is on [just-in-time design](#), producing just enough for development to start. The UX team itself is an agile team with different levels, including UX managers, UX product owners, researchers, strategists, and multiple interaction designers. As part of one agile team, the designers work with multiple SAFe development teams. The designers work ahead of development by two or three sprints and dedicate 10% of their capacity to design changes and integration support. Designers also work on certain research spikes in parallel.

The UX team develops a UX backlog and scopes and plans their work, as shown in Figure 10. Natalie Warnert published and popularized the UX Runway method and has detailed the process on her [Web site](#).



3.4 REFERENCES LINK

- [UXLifeCycle.pdf \(rit.edu\)](#)
- [The User Experience \(UX\) Lifecycle - Provis Media](#)
- [Choosing the Right UX Process for Your Software-Development Model :: UXmatters](#)
- [System Complexity Space - an overview | ScienceDirect Topics](#)
- [User Experience Team - Firesmith OPEN Process Framework \(OPF\) Website \(opfro.org\)](#)
- [UX — Scope \(Part 4\). Scope is the place where we translate... | by Omar El gabry | Omar El gabry's Blog | Medium](#)
- [The lifecycle approach to UX design | Inside Design Blog \(invisionapp.com\)](#)



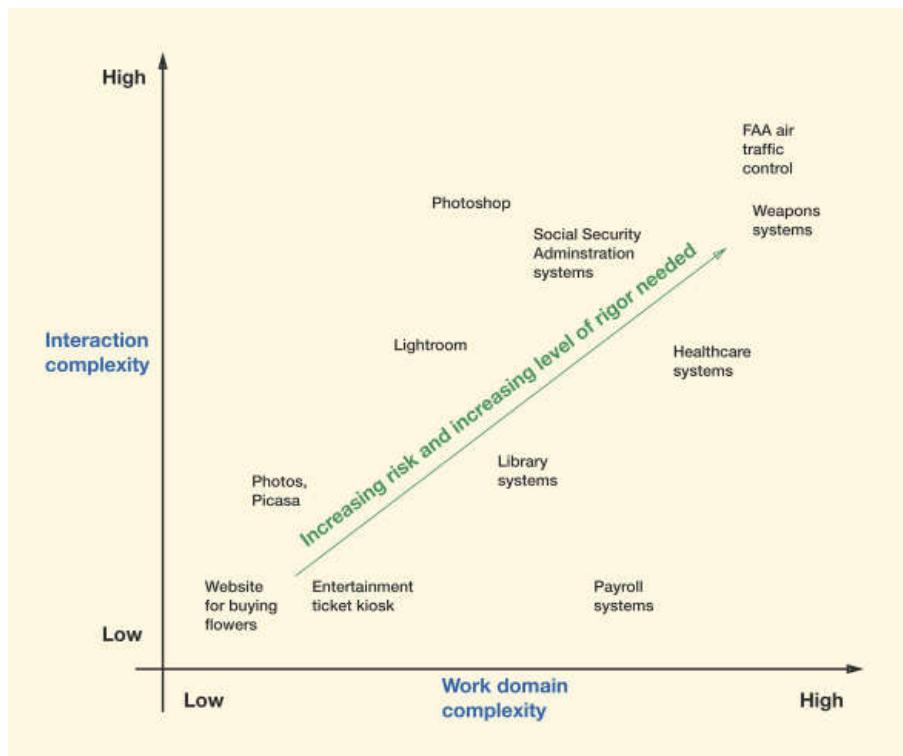
THE UX DESIGN – LIFE CYCLE-II

Unit Structure

- 4.1 The System Complexity Space
- 4.2 Meet The User Interface Team
- 4.3 Scope of UX Presence Within The Team
- 4.4 More About UX Lifecycles
- 4.5 References Link

4.1 THE SYSTEM COMPLEXITY SPACE

One big reason you can't define one set of methods for designing all systems is that there is a spectrum of system and product types with a broad range of risk versus needs for rigor in lifecycle activities and methods. In this and the next few sections, we look at some possibilities within this spectrum.



In Fig. 3-1, we show a system complexity space defined by the dimensions of interaction complexity and domain complexity (defined in the next two sections). While there undoubtedly are other ways to partition the space, this approach serves our purpose.

Interaction complexity

Interaction complexity, represented on the vertical axis, is about the intricacy or elaborateness of user actions, including the difficulty of cognitive actions, necessary to accomplish tasks with the system.

Low interaction complexity usually corresponds to systems that support smaller tasks that are generally easy to do, such as ordering flowers from a website. High interaction complexity is usually associated with larger and more difficult tasks, often requiring special skills or training, such as manipulating a color image with Adobe Photoshop (a high-functionality software application for managing and processing large collections of images and photographs).

Domain complexity

Ecology

In the setting of UX design, the ecology is the entire set of surrounding parts of the world, including networks, other users, devices, and information structures, with which a user, product, or system interacts .

On the horizontal axis, we show work domain complexity, which is about the degree of intricacy and the technical, or possibly esoteric, nature of the corresponding field of work. Convoluted and elaborate mechanisms for how parts of the system work and communicate within the ecology of the system contribute to domain complexity.

User work in domain-complex systems is often mediated and collaborative, with numerous “hand offs” in a complicated workflow containing multiple dependencies and communication channels, along with compliance rules, regulations, and exceptions in the way work cases are handled. Examples of high work-domain complexity include systems for geological fault analysis for earthquake prediction, global weather forecasting, and complete healthcare systems.

Low work-domain complexity means that the way the system works within its ecology is relatively simple. Examples of low domain complexity include that same website for buying flowers and a simple personal calendar management application.

The system complexity space quadrants

Simple interaction, simple work domain

The simplest quadrant is in the lower left corner of Fig. 3-1, where both interaction and work domain are the least complex. This quadrant contains smaller websites, certain interactive applications, and many commercial products. Just because this is the simple-simple quadrant, however, does not mean that the products are overly simple; the products of this quadrant can still be sophisticated.

There is an abundance of relatively simple systems in the world. Some, but not all, commercial software products are domain-simple and interaction-simple, at least relative to large systems of other types. Again, the website for ordering flowers is a good example of this quadrant. Interaction is very simple; just one main task involving a few choices and the job is done. Work domain complexity is also relatively simple because it involves only one user at a time and the workflow is almost trivial. Many apps on mobile devices, a significant segment of the commercial product market, are simple-interaction and simple-domain.

Although emotional impact factors do not apply to every system or product in this quadrant, emotional impact can be very important here, especially with respect to factors such as aesthetics or fun or joy of use. The smartphone and the personal mp3 music player are good examples of commercial products in this quadrant that have emotional impact issues, including meaningfulness (emotional impact within long-term usage).

Because systems and commercial products in this quadrant have less complexity, they generally require less rigor.

Complex interaction, complex work domain

As you move all the way across the diagonal in Fig. 3-1, you reach the upper right quadrant where you find interaction-complex and domain-complex systems, which are usually large and complicated.

Serious systems. Serious systems live in the far upper right corner of the system complexity space. An example is an air traffic control system used to decide the landing orders for an incoming airliner. An air traffic control system has enormous domain and interaction complexity, with workflow and collaboration among a large number of work roles and user types. Another defining example for this quadrant is a large system for the Social Security Administration.

For large domain-complex systems, such as military weapons systems, you are most likely to encounter resistance to innovation. Radical designs are not always welcome; conformity can be thought more important. Users and operators, in some cases, commit operations to habit and perform tasks with learned behavior even if there are better ways. Work practice change must be approached with caution.

This sector of serious systems within the system complexity space usually has little, if anything, to do with emotional impact factors such as aesthetics, fun, or joy of use.

Enterprise systems. Among the kinds of systems one sees in the rest of this quadrant are the so-called enterprise systems, large information systems used within organizations that have typically been forgotten in discussions of usability and user experience. Gillham (2014) puts it this way: “Most big businesses globally are locked into some kind of enterprise technology. Unfortunately, such systems are not only fiendishly difficult to install and maintain, but often equally challenging for the

workforce to use. When the stakes are so high, why is the user experience of enterprise systems so bad?"

The highest need for rigor. The domain-complex and interaction-complex quadrants are where you find the highest risks and, therefore, the highest need for rigor to manage risk. These project conditions include:

Risk

The danger or possibility of things going wrong, of features or requirements being missed, or the result not meeting the needs of users; possibility of not accommodating legacy needs or not complying with legal or safety constraints .

- When you have the greatest regulatory compliance requirements.
- When the importance of error avoidance in usage is highest (e.g., in mission-critical systems such as for air traffic control or for military weapons control).
- When you cannot risk getting these things wrong and the cost of failure is unacceptable.
- When you have a contractual obligation for high rigor.

Because of their sheer size and this need for rigorous processes, these systems are typically among the most difficult and expensive to design and develop.

Complex interaction, simple work domain

In the upper left quadrant of Fig. 3-1, one of the “in-between” quadrants, we see interaction-complex and domain-simple systems. It is typical of an interaction-complex system to have a large volume of functionality resulting in a large number and broad range of complex user tasks. The older-style digital watch (not a smart watch) is a lightweight but good example. Its interaction complexity often stems from a large variety of modal settings using overloaded and unlabeled push buttons. The domain, however, is still simple, being mainly about “what time is it?” Workflow is trivial; there is only one work role set within a simple system ecology.

Attention in this quadrant is needed for interaction design—myriad tasks, screen layouts, user actions, and even metaphors. Rigorous formative evaluation may be needed for the consistency and usability of the conceptual design and the detailed interaction design. The focus of modeling will be on tasks—task structure and task interaction models—and perhaps the artifact model, but not much attention will be given to work roles, workflow, or most of the other models .

For simple work domains, regardless of interaction complexity, usage research rarely results in learning something new that can make a difference in informing design. So less rigor will suffice leading up to design creation. For design creation, however, complex interaction

requires careful and systematic brainstorming, ideation, and sketching, plus iterative evaluation and refinement as well as attention to emotional impact factors.

Artifact model

A representation showing how users employ, manipulate, and share key tangible objects (physical or electronic work practice artifacts) as part of the flow in their work practice .

Simple interaction, complex work domain

In the lower right quadrant of Fig. 3-1, the other “in-between” quadrant, we see interaction-simple and domain-complex systems. In this quadrant, user tasks are relatively simple and easy to understand, so less attention to task descriptions is needed. The key effort for users in this quadrant is understanding the domain and its often esoteric work practice. Designers need rigorous usage research to focus on conceptual design and easy-to-understand user models of how the system works. Once that is understood, the interaction is relatively straightforward for users. Tax preparation software for average households is a good example because the underlying domain is complex but the data entry into forms can be simplified to a step-by-step process.

Sometimes systems for managing libraries, shown in the middle of the work domain complexity scale near the bottom of Fig. 3-1 fall into the simple-interaction, complex-work-domain quadrant. Typical library systems have low interaction complexity because the range of tasks and activities for any one user is fairly circumscribed and straightforward and the complexity of any one user task is low. Therefore, for a library system, for example, you do not need to exercise much rigor in the modeling of tasks.

However, a full library system has considerable domain complexity. The work practice of library systems can be esoteric and most UX designers will not be knowledgeable in this work domain. For example, special training is needed to handle the surprisingly important and highly controlled small details in cataloging procedures. Therefore, a rigorous approach to usage research may be warranted.

Gradations within the system complexity space

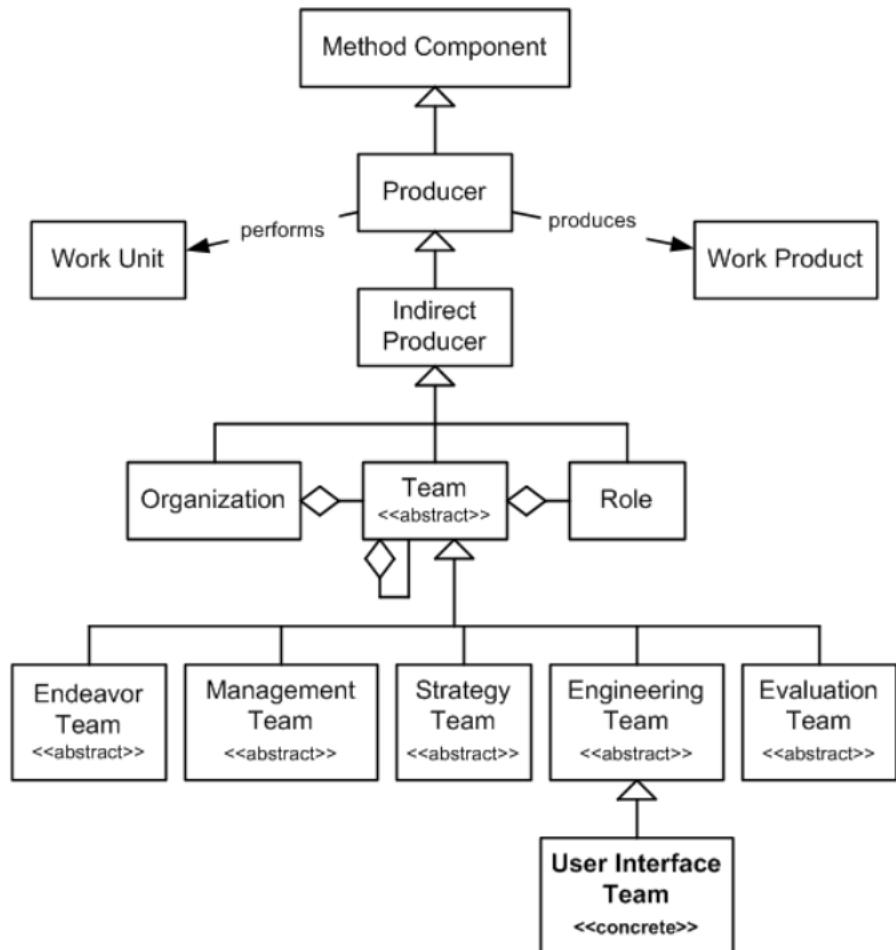
Some systems or products clearly fall into one quadrant or the other, but other projects can have fuzzy quadrant boundaries. Websites, for example, can belong to multiple quadrants, depending on whether they are for an intranet system for a large organization, a very large e-commerce site, or just a small site for sharing photographs. Products such as a printer or a camera are low in domain complexity but can have medium interaction complexity.

Healthcare systems typically cross system complexity space quadrants. Some parts internal to a small doctor’s office might be relatively simple.

But large healthcare systems that integrate medical instrumentation, health record databases, and patient accounting have complex work domains. Similarly, machines in a patient's hospital room can have a fairly broad range of technical tasks and activities, giving them relatively high interaction complexity.

The healthcare domain is also saddled with more than its share of regulation, paperwork, and compliance issues, plus legal and ethical requirements—all of which lead to high work domain complexity and a high need for rigor in UX lifecycle activities and methods.

4.2 MEET THE USER INTERFACE TEAM



Responsibilities

The typical team-specific responsibilities of a User Interface Team are to:

- Collaborate with the requirements team to engineer the usability requirements.
- Collaborate with the architecture team to architect the user interface.
- Design user interfaces that will:
- Meet user needs.
- Attract, engage, and retain users.
- Prototype and implement all user interfaces:
- Graphical User Interfaces (GUIs), such as websites
- Wireless User Interfaces
- Input Devices
- Output Devices
- Email and reports
- Perform usability testing.

A user interface team typically inherits the general team responsibilities from the team process component.

Roles

A user interface team typically consists of persons playing the following roles:

User Analyst, who analyzes the different user organizations and types of users and who collaborates with the requirements team to engineer the usability requirements

UI Designer,

who designs the user interfaces:

- GUI Designer, who designs the graphical (as well as textual and voice) user interfaces, who collaborates with the architecture team to produce the navigation diagram, and who collaborates with the digital branding team with regard to the digital branding of the interface.
- Hardware Designer, who designs the hardware user interfaces (e.g., physical buttons, scanners).

User Experience Design

Content Creator,

who collaborates with the content management team to create content for the user interface:

- Audio Artist
- Content Author
- Graphic Artist
- Video Artist

Hardware Engineer,
who fabricates the hardware user interfaces

Usability Engineer,
who programs the software user interfaces

Test Engineer,
who collaborates with the independent test team to develop the usability tests

Test Monitor,
who runs the usability tests

Technical Writer,
who documents the user interfaces

Tasks

A user interface team performs the following team-specific design, implementation, and testing tasks in an iterative, incremental, parallel, and time-boxed manner:

- Requirements Engineering of the User Interfaces:
 - User Analysis
 - Requirements Identification
 - Requirements Analysis
 - Requirements Specification
- Architecting of the User Interfaces:
 - Architecture Production
 - Architecture Documentation via navigation diagrams
- Design of the User Interfaces:
 - Design Production
 - Design Prototyping

- Design Documentation
- Design Refactoring
- Implementation of the User Interfaces:
 - Data Generation of user interface data
 - Fabrication of hardware user interfaces
 - Programming of software user interfaces
 - Debugging of user interfaces
- System Usability Testing:
 - Test Reuse
 - Test Design
 - Test Implementation
 - Test Execution
 - Test Reporting

A user interface team typically inherits the common team tasks from the team process component.

Guidelines

- This method component (process component class) is typically tailored via subclassing to add, modify, and remove responsibilities, roles, tasks, and work products.
- Clarify and minimize any overlap in responsibilities between the user experience team and the:
 - Digital branding team with regard to the usability of the digital brand.
 - Requirements team with regard to engineering the usability requirements.
 - Architecture team with regard to overall user interface architecture and design.
 - Content management team with regard to data component (content) production.
 - Software development team with regard to the integration of user interface software components with the other software components.
 - Independent test team with regard to usability testing.

- Some development organizations consider user experience a discipline rather than a team. Instead of constructing a user experience team, such organizations instead distribute the above roles to the above teams. This helps avoid “turf battles” between these teams.
- A user experience team typically inherits the common team guidelines from the team process component.

4.3 SCOPE OF UX PRESENCE WITHIN THE TEAM

Scope is the place where we translate user needs and business objectives into specific requirements for what content and functionality the product will offer to users.

Why to both yourself with defining the Scope

Know what we are building

It gives a clear description about what we are building, and how it related to our goals. and Yes, we can't define a scope with 100%, and say this is exactly what is going to be, because new things are going to come during the project, but, if we defined boundaries of what you can and can't do in the Scope, it will be much easier.

Address conflicts

Defining scope makes us to address potential conflicts, and the bottlenecks from implementing the features before design and development.

Reference point

Defining scope gives a reference point for client and developers about the work that's going to be done. Documentation of the Scope doesn't have to be a novel, but a common understanding of the features, schedules, and milestones puts everybody on the same page, and gives them a common reference.

It also gives a common language for talking about that work, like when you say “job vacancies search form”, now everybody knows what you are talking about.

Explore connections

Defining scope gives enables you to see connections between individual requirements that wasn't clear, and organize related requirements, which may help then in determining which structure is appropriate for your product.

Assign Responsibility

Having a defined set of requirements allows you to assign responsibility for the work, so no body will have a conflict about who is responsible for what tasks.

Keep track of new features

More features are coming during the project. Because there is information coming to the light and So, suggested changes or new features should also be documented.

What to tackle now and what has to wait

Having a list with set of features to be implemented can help in determine which features can be done now and which has to wait until the next iteration.

Functional and Content Requirements

We start by generating the requirements, negotiate, prioritize and finally we document the requirements.

Generating Requirements

Some requirements apply to the product as a whole, such as supported browsers and operating systems, while other requirements apply only to a specific feature. The level of detail in your requirements will often depend on the specific scope of the project. The complex, the more detailed.

The source for requirements will be the users along with the stakeholders. The techniques used in the Strategy for research can be used here to help in generating the requirements. The requirements that will be generated will fall into three categories:

Things people say they want

Either people try to remember their past use for the product, or speculate about their future use. And the problem is, people make false predictions about their future behavior, especially when presented with something that's new and unfamiliar.

Things people aren't actually want

Sometimes the things people say they want are not the things they actually want, because when people have troubles with a process or a product, they try to imagine a solution, but not all solutions are feasible.

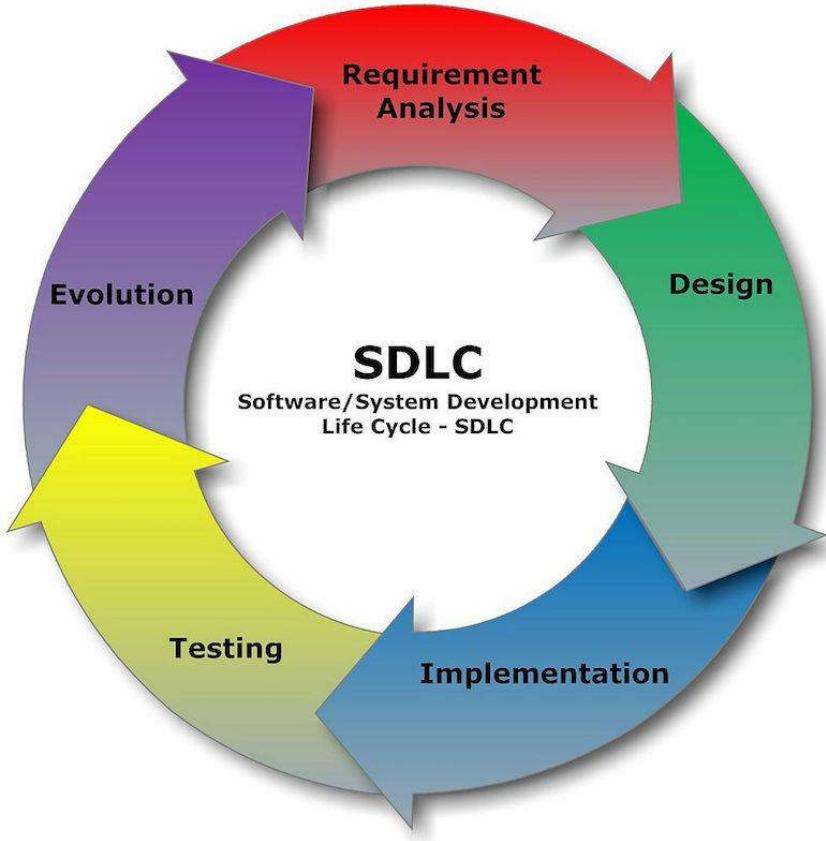
Things people don't know they want

When people talk, they may hit upon great ideas that nobody said before. These ideas often come from brainstorming sessions.

4.4 MORE ABOUT UX LIFECYCLES

Much has been said of the agile lifecycle. We all know it: discover, design, develop, test, iterate, repeat.

Some project managers even repeat the process in their sleep. There are tons of well-known reasons to run an agile lifecycle on a project.



The most obvious reason: efficiency. Every leg of the journey is clearly defined, and each member of the team knows their role. The caveat? User experience bleeds into every part of the process, unbeknownst to many patrons of the profession.

User experience is a part of everything

For the skeptics who think that development is independent of UX, here's the thing: the work of a UX designer is almost never complete—*especially* once the project enters the development phase.

Visual designers, developers, and QA specialists exist solely to realize the vision of the UX designer. So it's imperative that UX designers maintain an active role in every forthcoming leg of the project lifecycle.

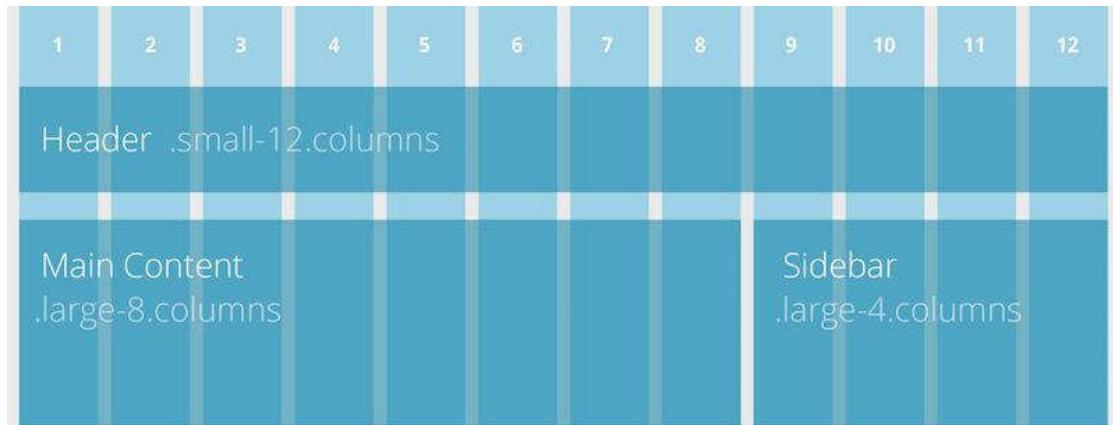
“User experience is a part of everything.”

If you're asking yourself why it's the UX designer's responsibility to ensure that everyone else does their job, you're asking the right question. Let me explain why.

Development choices have an effect on user experience

There are a number of choices that a developer can make as they work on a project that may adversely affect the usability of a website. These choices can relate to performance, programming language preference, and accessibility, to name a few.

If, for example, the developers have opted to use a front-end framework with a number of default grids, such as Foundation by Zurb, then it's imperative that they match the the usability requirements determined by the UX designer.



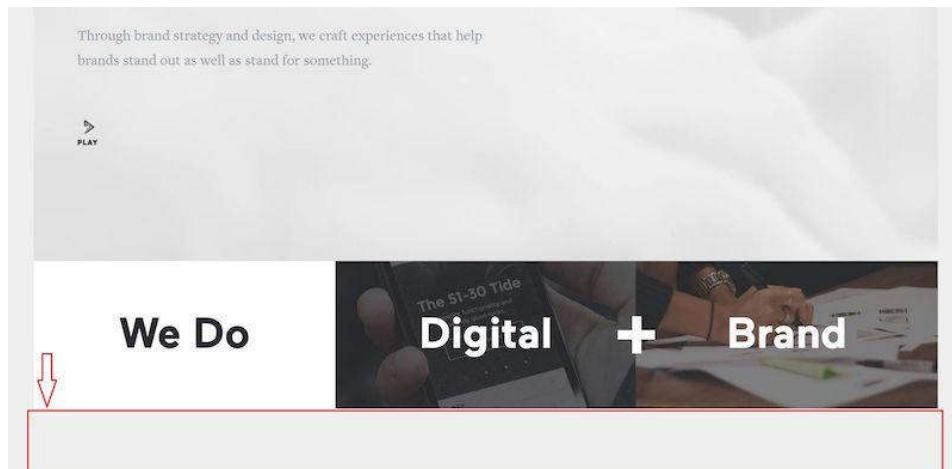
Zurb Foundation Grid.

In the above image, the sidebar has been allocated columns 9-12. It seems like a natural choice for the developer, but it could pose a number of usability challenges that only a UX designer would catch.

Other decisions, like ones that affect performance, also affect usability. So it's up to the UX designer to ensure that any decisions that affect performance stay within the realm of their vision.

If a page fails to load within the time deemed acceptable by the UX designer, then it's not usable. Deeper still, if particular *assets*, such as images and dynamic animations don't function/load smoothly, they can also be a detriment to usability.

On the other hand, an implementation such as lazy loading, which aims to help pages load more quickly, can also negatively impact the usability of a page. Whether it does can only be determined by the visionary UX designer.



In the above example of an agency website with lazy loading, check out how the elements under “We Do Digital + Brand” haven’t loaded yet, even though the user has scrolled a good deal below it.

If this is consistent with the vision of the UX designer, then lazy loading would be an example of a development choice being beneficial.

But if the UX designer is less than excited about the fact that it takes an additional second for every new element to load as you scroll, then the implementation can be deemed detrimental to the usability of the page.

Collaboration among teams is vital

How does the UX designer ensure that the finished product is as close to their vision as possible? The key is collaboration. Effective collaboration is not only mission-critical, but also well within reach—even for the busiest of UX designers.

There are 3 ways to achieve effective collaboration:

1. Set up communication waypoints Waypoints are predetermined times at which the primary UX designer ought to meet with the development team to make sure that the choices they have made are within the scope of the UX vision.

How to do it First, determine the length of each sprint, and the amount of time it’ll take for the development team to build out each module. This information can usually be acquired from the project manager, who is used to working with timelines and feature build-outs.

Next, establish a review calendar of sorts, and set specific meeting times so that you can examine the work that’s been done on the project so far. Every time you meet with the development team, find out what features they’ve built out and the potential limitations of their work.

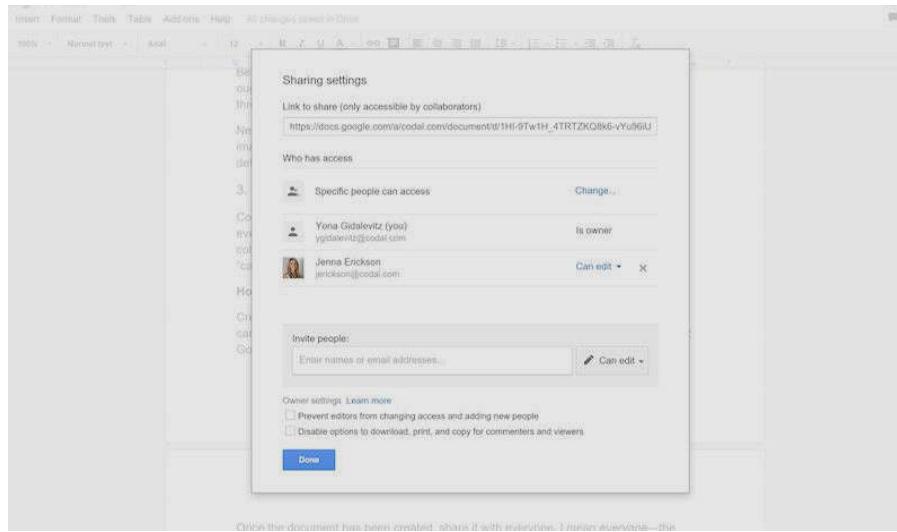
2. Define any benchmarks Benchmarks are descriptors, both quantitative and qualitative, which aid in the implementation of a particular vision. Because benchmarks can be met in a variety of ways, UX designers should review benchmarks at each waypoint meeting.

How to do it Before you can present the development team with usability benchmarks, you have to first determine *what* needs to be benchmarked. This can be done as you go through the process of crafting the user experience.

Next, you have to determine *how* each benchmark will be measured. If there are image-heavy elements above the fold that are essential to the experience, define an acceptable load time for the page, and take note of it for yourself.

3. Use something like Google Drive Collaborative workspaces like Google Drive are *extremely* effective in keeping everybody up-to-date

with everyone else's work. Because of the way they work, collaborative tools can eliminate miscommunication and reduce the amount of "catching up" that needs to be done during the waypoint meetings.



How to do it Create a master document for all relevant information, then share it with everyone. I mean *everyone*—the developers, the UX team, the project manager, even the client(s). The result is total transparency for everyone involved in the project.

Take charge

UX designers: don't be afraid to be decisive and tell the developers when they aren't adhering to the usability guidelines made available to them. Involve yourself in every leg of the software development lifecycle, and remain vigilant—you never know when a developer will do something *without* documenting it.

4.5 REFERENCES LINK

- [UXLifeCycle.pdf \(rit.edu\)](#)
- [The User Experience \(UX\) Lifecycle - Provis Media](#)
- [Choosing the Right UX Process for Your Software-Development Model :: UXmatters](#)
- [System Complexity Space - an overview | ScienceDirect Topics](#)
- [User Experience Team - Firesmith OPEN Process Framework \(OPF\) Website \(opfro.org\)](#)
- [UX — Scope \(Part 4\). Scope is the place where we translate... | by Omar Elgabry | OmarElgabry's Blog | Medium](#)
- [The lifecycle approach to UX design | Inside Design Blog \(invisionapp.com\)](#)



THE UX DESIGN PROCESS – UNDERSTAND USERS

Unit Structure

- 5.0 Objectives
 - 5.1 Introduction
 - 5.2 The system Concept Statement
 - 5.3 User work activity gathering
 - 5.4 Look for emotional aspects of work practice
-

5.0 OBJECTIVES

A UX research objective is **a statement of what you want to learn about your customers (or users) from carrying out the research**. But that is wrong, or at least not accurate enough. A well written list of research objectives: Delineate and align with the goal of the project

5.1 INTRODUCTION

What is UX design?

User experience (UX) design is the process of building products that are easy and enjoyable for people to use. By building products to be as effortless as possible for users, you can encourage adoption, retention, and loyalty.

UX is an iterative process of constant improvement: product teams and designers use data and usability testing to continuously refine the product experience so it becomes easier for users as the product develops.

The UX design process in 5 steps

There's no universal workflow that every UX and product team follows—products, users, and internal processes vary. Some teams follow the Lean UX model, for example, which includes three broad phases:

1. **THINK:** teams draw on user feedback, product and usage data, competitive analysis, and other research to identify blockers and pain points, and brainstorm ways to improve the product to solve problems for their users.
2. **MAKE:** developers and designers build the change or new feature into the product.

3. **CHECK:** teams use surveys, A/B or multivariate (MVT) testing, and other tools and methods to check whether the change or new feature improves UX and solves the original problem for their users.

The basic UX design process we outline below is similar, but we break it down into five steps to illustrate the **cyclical, iterative nature of UX design:**

1. Research and understanding

User experience design begins with extensive UX research before any designs are mocked up.

- **User research:** identify pain points, goals, blockers, and characteristics among users, and draw up detailed user personas that inform design decisions to keep the UX design process focused on users.
- **Market research:** understand the market and similar products available, what those products share in terms of UX, and identify opportunities to differentiate your product based on UX.
- **Historical analysis:** get a sense of the history of your product, and use previous iterations to inform future design decisions, ensure mistakes aren't repeated, and continuously improve your product's UX based on what you've learned.

2. Prototyping and wireframing

After the research stage, a designer creates a prototype and wireframe of your product, outlining how users will flow from one feature or element to the next, and how the product will look visually.

The prototype is your minimum viable product (MVP): a version of your product with enough features to allow your first users to test it and offer feedback. The prototype generally doesn't look like a finished product, but it provides a sense of how the finished product will work.

Prototyping and wireframing—and then testing, which is the next step—lets you get valuable user feedback *before* you spend resources finalizing your product, potentially avoiding a lot of work and time spent developing the wrong features and UX for your users.

3. Testing

Next, your product team will conduct user tests—like A/B tests, MVT, and split tests—to **identify existing or potential design issues** that may cause friction for users. For example, your team may find:

- The UX is confusing and prevents users from accomplishing their goals
- Some features aren't worth developing further because users aren't interested in them

- Certain user actions create friction in the product experience

Some user testing may include using product experience and behavior analytics tools like heatmaps and session recordings and collecting user feedback that can help identify blockers and pain points.

4. Launch

Throughout the testing phase, you'll refine your prototype based on results and user feedback, make changes, and add final touches before launch.

Once your product is ready, **launch is often done in stages**. For example, your UX team might release a beta version or ‘soft launch’, where a new design is launched to a small group of users first, to get additional user feedback and usage data before rolling it out to your whole user base.

5. Iteration

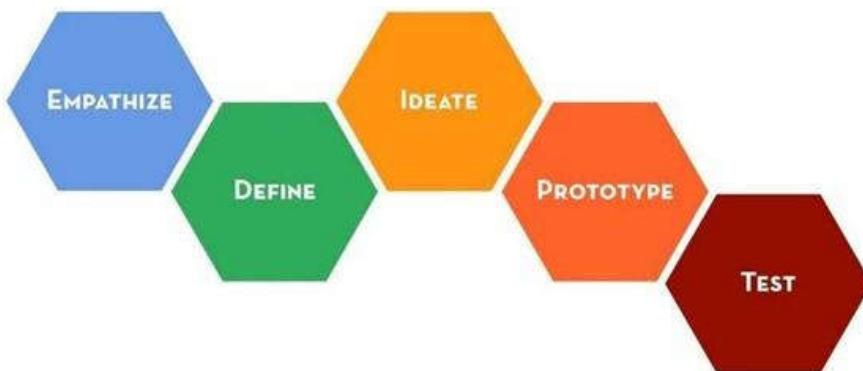
UX design is a constant, iterative process—you’re never *done* creating good UX—so more testing follows the product’s launch.

In the iteration stage, your team will continue to make updates and improvements as needed, and closely monitor how those changes impact the product experience.

A/B and multivariate testing are often part of this stage, so your team can see the impact of product changes and compare multiple versions of the design to see which creates the best user experience. Then the winning version is rolled out.

5.2 THE SYSTEM CONCEPT STATEMENT

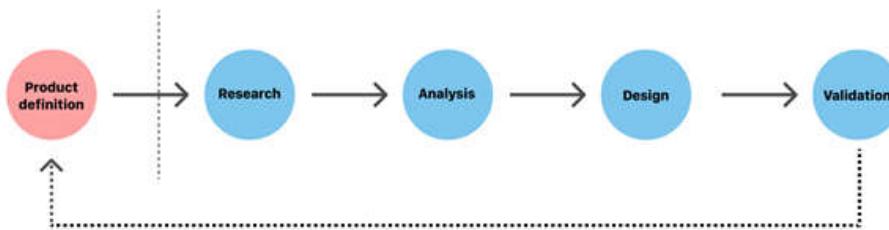
Most designers are familiar with the concept of “design thinking” as a UX process. This process has five stages in it: empathize, define, ideate, prototype, and test. Most design processes originate from this concept.



The “design thinking” process, defined by the Stanford School of Design, includes five stages: empathize, define, ideate, prototype, and test. Image credit Hasso Plattner Institute of Design.

If we apply design thinking to product design, we would follow a UX process with the following five key phases:

- Product definition
- Research
- Analysis
- Design
- Validation



The UX design process consists of five key phases: product definition, research, analysis, design, and validation. Image credit Nick Babich.

1. Product definition

One of the most important phases in UX design is actually done before the product team creates anything. Before you can build a product, you need to understand its *context for existence*. The product definition phase sets the foundation for the final product. During this phase, UX designers brainstorm around the product at the highest level (basically, the concept of the product) with stakeholders.

This phase usually includes:

- **Stakeholder interviews:** interviewing key stakeholders to gather insights about business goals.
- **Value proposition mapping:** thinking about the key aspects and value propositions of the product: what it is, who will use it, and why they will use it. Value propositions help the team and stakeholders create consensus around what the product will be and how to match user and business needs.
- **Concept sketching:** creating an early mockup of the future product (can be low-fidelity paper sketches of the product's architecture).

This phase typically ends up with a project kick-off meeting. The kick-off meeting brings all the key players together to set proper expectations both for the product team and stakeholders. It covers the high-level outline of the product purpose, team structure (who will design and

develop the product), communication channels (how they will work together), and what stakeholders' expectations are (such as KPIs and how to measure the success of the product).

2. Product research

Once you've defined your idea, the product team moves to the research phase. This phase typically includes both user research and market research. Seasoned product designers think of research as a good investment—good research informs design decisions and investing in research early in the process can save a lot of time and money down the road.

The product research phase is probably the most variable between projects—it depends on the complexity of the product, timing, available resources, and many other factors. This phase can include:

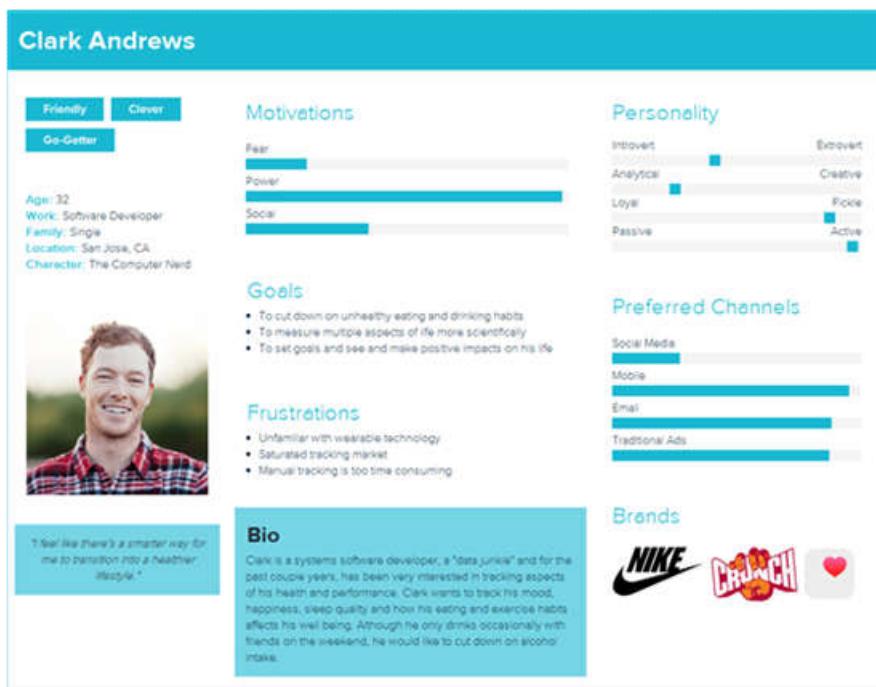
- **Individual in-depth interviews (IDI).** A great product experience starts with a good understanding of the users. In-depth interviews provide qualitative data about the target audience, such as their needs, wants, fears, motivations, and behavior.
- **Competitive research.** Research helps UX designers understand industry standards and identify opportunities for the product within its particular niche.

3. Analysis

The aim of the analysis phase is to draw insights from data collected during the research phase, moving from “what” users want/think/need to “why” they want/think/need it. During this phase, designers confirm that the team’s most important assumptions are correct.

This phase of the UX process usually includes:

- **Creating user personas.** Personas are fictional characters that represent the different user types for your product. As you design your product, you can reference these personas as realistic representations of your target audience.



Example of a user persona, showcasing the person's gender, age, motivations, and more. Image credit Xtentis.

- **Creating user stories.** A user story is a tool that helps designers understand the product/service interactions from the user's point of view. It's usually defined with the following structure: "*As a [user] I want to [goal to achieve] so that [motivation].*"
- **Storyboarding.** Storyboarding is a tool that helps designers *connect* user personas and user stories. As the name suggests, it's essentially a story about a user interacting with your product.

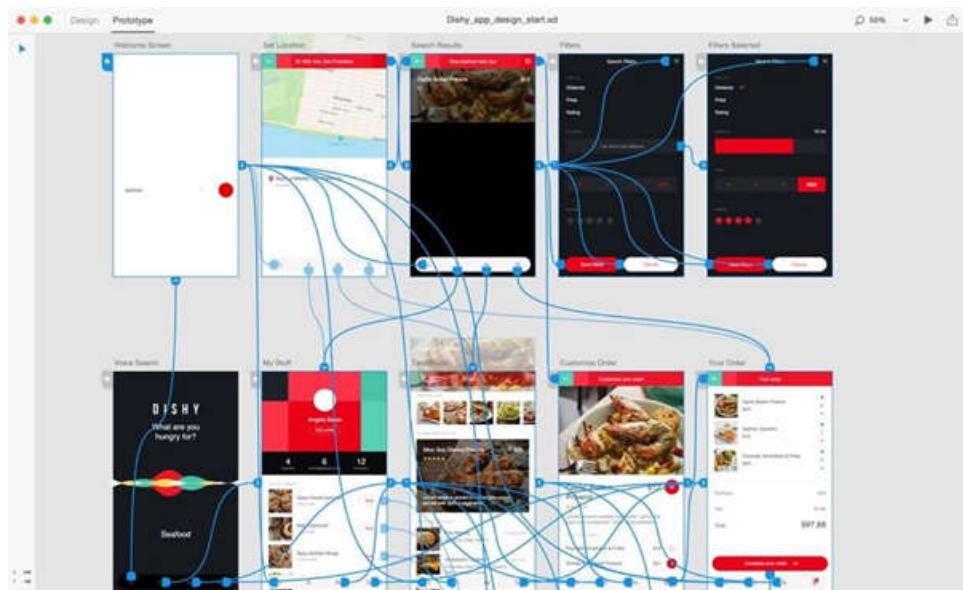
4. Design

When users' wants, needs, and expectations from a product are clear, product designers move to the design phase. At this step, product teams work on various activities, from creating information architecture (IA) to the actual UI design. An effective design phase is both highly collaborative (it requires active participation from all team players involved in product design) and iterative (meaning that it cycles back upon itself to validate ideas).

The design phase usually includes:

- **Sketching.** Sketching is the easiest and fastest way to visualize our ideas. You can do this by drawing by hand on a piece of paper, on a whiteboard, or in a digital tool. It's very useful during brainstorming sessions because it can help the team visualize a broad range of design solutions before deciding which one to go with.

- **Creating wireframes.** A wireframe is a tool that helps designers visualize the basic structure of a future page, including the key elements and how they fit together. Wireframing acts as the backbone of the product, and designers often use them as a foundation for mockups and prototypes.
- **Creating prototypes.** While wireframes are mostly about structure and visual hierarchy (the look), prototypes are about the actual interaction experience (the look and feel). A prototype is like a simulation of the product and may be low-fidelity (clickable wireframes) to high-fidelity (coded prototypes).



With Adobe XD, you can turn static mockups into a prototype by creating a connection between individual screens. Image credit Adobe XD.

- **Creating a design specification.** Design specifications contain all of the visual design assets required for developers to turn prototypes into a working product.
- **Creating design systems.** For large projects, designers typically create a system of components, patterns, and styles that help both designers and developers stay on the same page regarding the design.

5. Validation (Testing)

Validation is an essential step in the design process because it helps teams understand whether their design works for their users. Usually, the validation phase starts after the high-fidelity design is ready, since testing with high-fidelity designs provides more valuable feedback from end-users. During a series of user testing sessions, the team validates the product with both stakeholders and end-users.

The validation phase of the UX process may include the following activities:

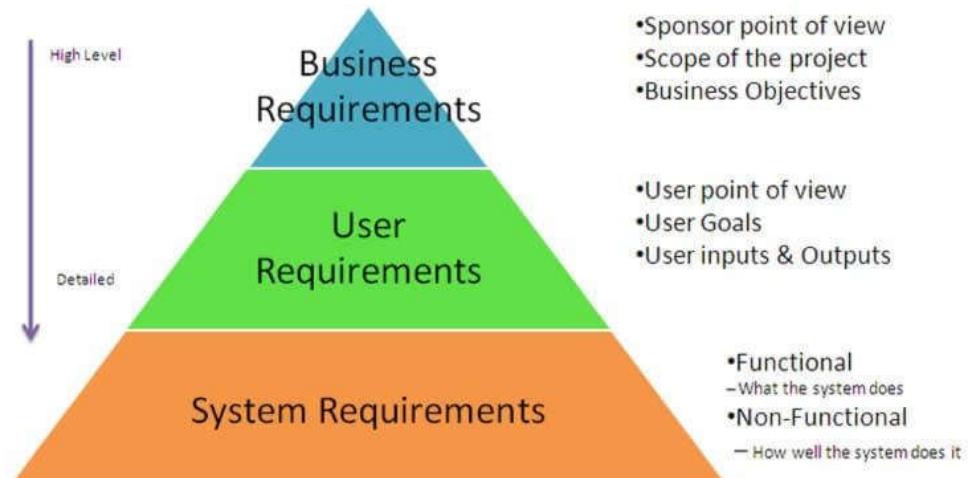
- **“Eat your own dogfood.”** Once the design team has iterated the product to the point where it’s usable, it’s time to test the product in-house. Team members should try using the product on a regular basis, completing routine operations to uncover any major usability flaws.
- **Testing sessions.** User testing sessions with people who represent your target audience are very important. There are many different formats to try, including moderated/unmoderated usability testing, focus groups, beta testing, and A/B testing.
- **Surveys.** Surveys are a great tool for capturing both quantitative and qualitative information from real-world users. UX designers can add open-ended questions like “What part of the product you dislike?” to get user opinions on specific features.
- **Analytics.** Quantitative data (clicks, navigation time, search queries, etc.) from an analytics tool can be very helpful to uncover how users interact with your product.

5.3 USER WORK ACTIVITY GATHERING

A great user experience is all about enabling users achieve their objective when using your *artifact* – be it a website, a software system or anything that you create. Now take a step back. Trying to understand how to make it easy for users to achieve their goals would be pointless if you don’t place it within the context of what you know about your users. The more you understand your users, their work and the context of their work, the more you can support them in achieving their goals – and hence, the more usable your system will be! So, you inevitably ask the question “how would I know what my users’ needs are?” This article is about requirements gathering … and it answers this question.

What are Requirements?

A **requirement** is a statement about an intended product that specifies what it should do or how to do it. For requirements to be effectively implemented and measured, they must be specific, unambiguous and clear. For example, a requirement may be that a specific button must enable printing of the contents of the current screen.



Diagrammatic representation of the different types of requirements
(Source: SatheesPractice)

Since this article focuses on requirements gathering of systems, we will focus on the two types of **System Requirements**:

1. Functional Requirements

Functional requirements specify the software functionality that the developers must build into the product to enable users to accomplish their tasks, thereby satisfying the business requirements. In simpler words, functional requirements state what the system must do. In fact, functional requirements are usually stated by using the “shall” statement. For example, “the website shall notify the administrator via email when a user registers with it”

- Business rules
- Transaction corrections, adjustments
- Administrative functions
- Authentication
- Audit tracking
- External interfaces
- Certification requirements
- Reporting requirements
- Historical data
- Legal / Regulatory requirements

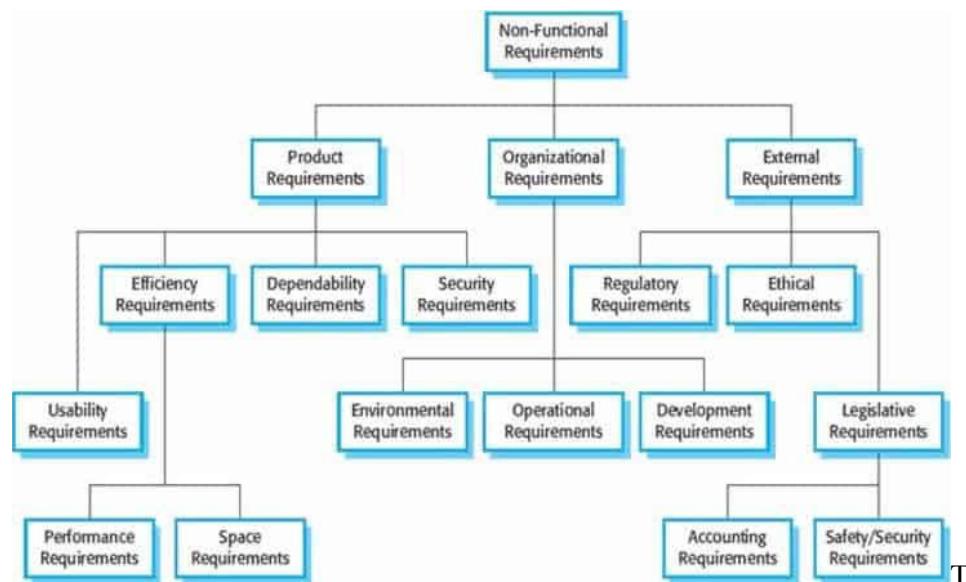
2. Non-Functional Requirements

Constraints or standards that the system must have or comply with. Non-functional requirements define the system's quality characteristics. As a

rule of thumb, non-functional requirements generally end with “ity”, although not all of them do.

The UX Design Process –
Understand Users

- Scalability
- Capacity
- Availability
- Reliability
- Recoverability
- Maintainability
- Serviceability
- Security
- Regulatory
- Manageability
- Environmental
- Data Integrity
- Usability
- Interoperability
- Performance



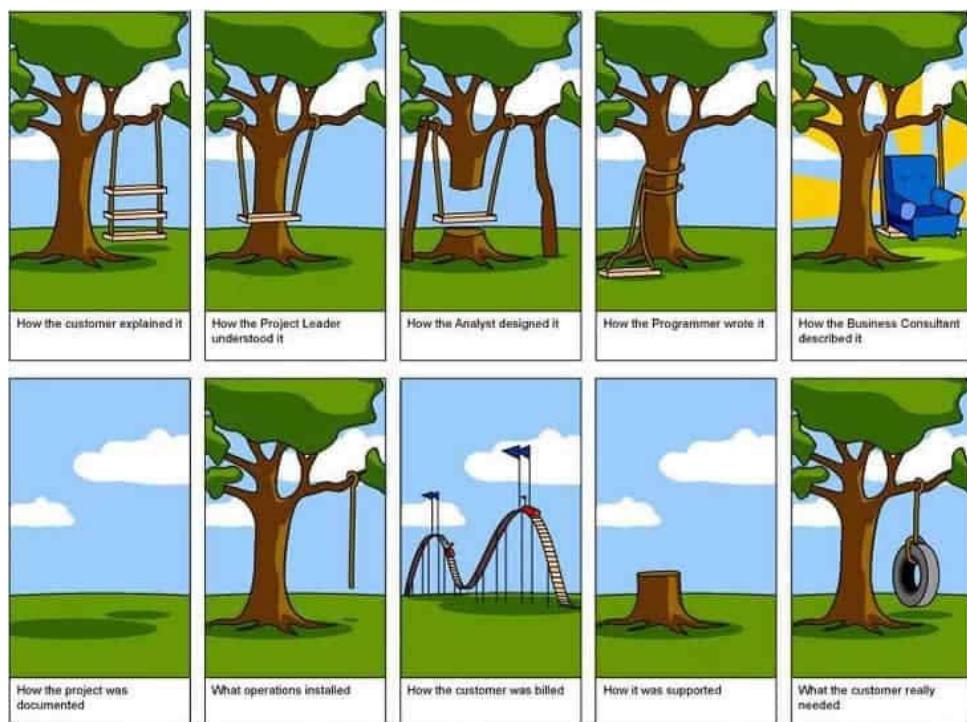
This diagrammatic representation of non-functional requirements shows how they are related (Source: Robinsce)

The Iterative Nature of Requirements Gathering

In an ideal world, one would simply gather data related to User needs, analyse it and then elicit the user requirements. However, this is a very simplistic view. In the real world, user requirement gathering is an iterative process whereby each of the above steps influences the other. For example, when trying to set a particular user requirement, you realise that it is not very clear if the user really wants what you think they want. Therefore, you may opt to gather more data as a means to clarify this ambiguity. In addition to this, you will realise that the requirements themselves evolve as stakeholders interact with the prototypes that you develop based on your initial requirements gathering. What follows is a practical 3-step approach on how to gather data from your users and convert this data into system requirements.

Step 1: Gather Data From The Users

At this early stage, do not restrict your definition of users to the actual users of your system. Instead, widen it to include a sample that represents each stakeholder.



The famous cartoon entitled “Project Cartoon” captures the sad reality behind bad requirements gathering (Source: Project Cartoon)

Very often you will find that actual users will help you formulate functional requirements. However, the non-functional requirement will usually come from other stakeholder groups such as the people who are providing the funding of the system. For example, financial controllers would be interested in the scalability of the system since if the system is scalable, then adding more functionalities to it will not involve throwing away the current system.

According to Jenny Preece and Helen Sharp, in their book *Interaction Design: Beyond Human-Computer Interaction*, data gathering can be done using the following conventional techniques:

- **Interviews:** Interviews are good for getting people to explore issues. Semi-structured or unstructured interviews are often used early on to elicit scenarios. In the context of establishing requirements, it is equally important for development team members to meet stakeholders and for users to feel involved. This on its own may be sufficient *motivation* to arrange interviews.
- **Focus Groups:** Focus groups are ideal for establishing a consensus view and highlighting areas of conflict and disagreement during the requirements activity. On a social level it also helps for stakeholders to meet designers and each other, and to express their views in public. It is not uncommon for one set of stakeholders to be unaware that their understanding of an issue or a process is different from another's even though they are in the same organization.
- **Questionnaires:** Questionnaires may be used for getting initial responses that can then be analyzed to choose people to interview or to get a wider perspective on particular issues that have arisen elsewhere. Or the questionnaire might be used to get opinions and views about specific suggestions for the kind of help that would be most appreciated.
- **Direct Observation:** Observation of participants in their natural setting is used to understand the nature of the tasks and the context in which they are performed. Sometimes the observation is carried out by trained observers who record their findings and report them back to the design team, and sometimes the observation is carried out by or with a member of the design team.
- **Indirect Observation:** Diaries and interaction logging are used less often within the requirements activity. Interaction logging on an existing system may be used to provide some data about how a task is performed currently, but the information is too tightly coupled with details of the existing computer support to be particularly helpful if a completely new system is planned.
- **Studying Documentation:** Manuals and other documentation are a good source of data about the steps involved in an activity and any regulations governing a task. Such documentation should not be used as the only source, however, as everyday practices may augment them and may have been devised by those concerned to make the procedures work in a practical setting. Taking a user-centred view of development means that we are interested in the everyday practices rather than an idealized account.
- **Researching Similar Products:** By observing and analyzing similar products, it is very easy to establish the requirements of your own product
-

5.4 LOOK FOR EMOTIONAL ASPECTS OF WORK PRACTICE

The most common emotions at work

To create awesome emotional cultures, organizations need to understand which emotions are prevalent in the workplace and how they affect employees.

Here are the most common positive and negative emotions at work:

1. Comfortable
2. Satisfied
3. Enthusiastic
4. Frustrated
5. Stressed
6. Anxious

Positive emotions at work

Positive emotions aren't limited to only optimistic and hopeful feelings. Examples of positive emotions could include calm, comfortable, energetic, enthusiastic, excited, happy, joyful, peaceful, relaxed, and satisfied.

Our research found that the top three positive emotions felt by employees at work are:

- Comfortable (47.8%)
- Satisfied (37.1%)
- Enthusiastic (36.6%)

Negative emotions at work

There are a lot of negative emotions that can surface at work. Examples of negative emotions could include annoyed, anxious, bored, disinterested, dissatisfied, frustrated, gloomy, miserable, sad, stressed, tired, uncomfortable, unhappy, upset, and worried.

Here's what employees identified as the top three negative emotions felt at work:

- Frustrated (56.2%)
- Stressed (45.1%)
- Anxious (30.4%)

Negative moral emotions at work

Basic emotions like happiness and sadness are pretty straightforward and easy to understand. But many of the emotions we experience in the workplace are much more complex—especially moral emotions.

Morals are the principles that differentiate between good and bad behavior, and they vary from person to person. Emotions that are

influenced by our morals are called moral emotions. They include emotions like guilt, regret, and shame—and they carry much more weight than our primary emotions.

When taking average rank and top 3 percentage into account, employees believe the most negative moral emotions are:

- Humiliation
- Disgust
- Resentment

How emotions differ throughout the organization?

There's a lot that can contribute to both negative and positive emotions at work. Many factors occur in our personal and professional lives, but what about things like tenure and position level?

Individual contributors may feel frustrated from a lack of support or satisfied when they receive a promotion.

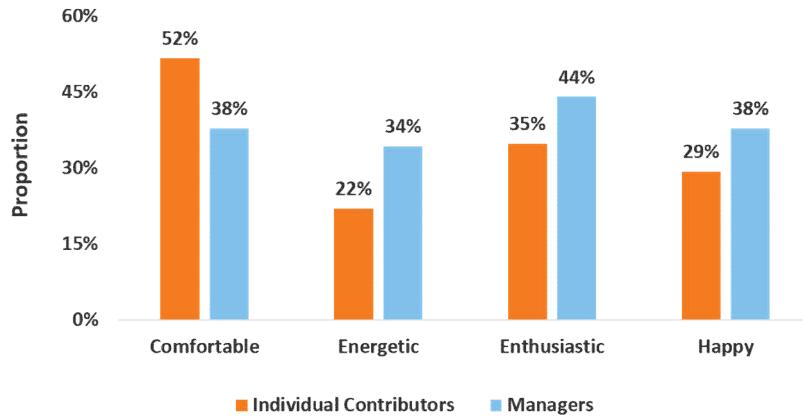
Managers may feel stressed from conducting many 1:1s or comfortable with the progress their team has made.

Leaders may feel anxious about the forecast of their business or enthusiastic about starting the year fresh with a brand new focus.

We are all capable of feeling a variety of emotions at work. Yet, our research uncovered some differences between individual contributors and managers.

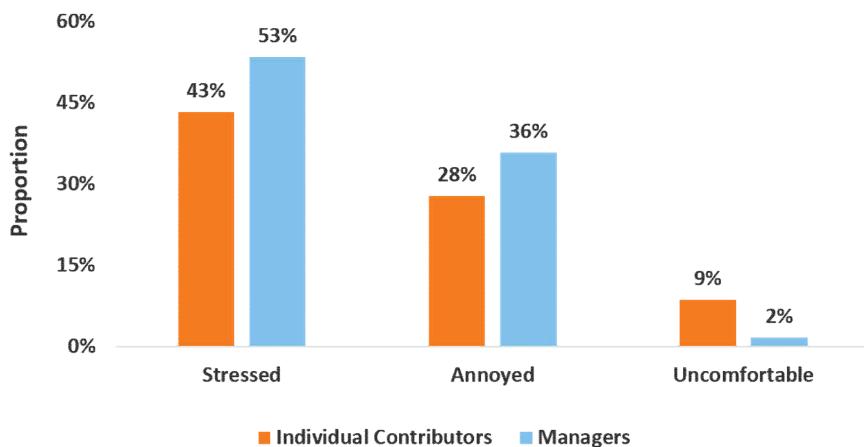
Managers are more likely to feel energetic, enthusiastic, and happy—but managers likely feel comfortable less often.

Differences in Positive Emotions by Position Level



Managers are more likely to feel stressed and annoyed—but managers likely feel uncomfortable less often.

Differences in Negative Emotions by Position Level



There are many factors that contribute to our emotions. Managers may tend to feel positive emotions more often due to their roles as coaches. At the same time, managers often have more responsibilities and encounter more barriers than individual contributors. This could explain a higher frequency of stress.

The relationship between emotions and employee engagement

For some, emotions can take a toll on our day. Some may disengage from their work and some may miss work altogether. Our research shows some clear connections between emotional culture and levels of engagement. Engagement impacts a wide variety of important business outcomes—making a solid case for addressing emotions at work.

Burying emotions hurts engagement, but so does being in a toxic environment. When the people around you are frequently and openly expressing negative emotions, it can have a damaging effect on employee engagement.

Employee engagement and negative emotions

Experiencing negative emotions expressed by others may decrease feelings of connection. As employee engagement decreases, employees may become more sensitive to negative emotions.

To help decrease the expression of negative emotions at work, you need to understand what happened before those emotions boiled over. Each type of emotion is unique and requires a unique approach to handling it.

Employee engagement and emotional outcomes

In theory, highly engaged employees are more open-minded when it comes to potential outcomes of negative emotions. Yet, they are more close-minded when it comes to potential outcomes of positive emotions.

On one hand, engagement is a negativity filter. At the same time, it promotes positivity bias. Bottom line—highly engaged employees are more positive than less engaged employees.

But it's also possible that disengaged employees have a harder time coping with emotions in healthy ways. Consider evolving your approach to disengaged employees by viewing them through the lens of reconnection. Help them strengthen their coping mechanisms to handle stress at work.

How to manage emotions in the workplace?

When left unchecked, employee emotions can have a serious impact on the workplace. Make sure you understand each of these unique emotions and strive to find the right balance of supporting and preventing them at work.

Building a culture of trust and practicing emotional intelligence skills can allow employees to acknowledge, feel, and express more positive emotions in the workplace. Here are some ways to manage emotions in the workplace.

Coping mechanisms for employee emotions

Coping mechanisms are the tools and strategies we use to deal with stress in our lives. Our various ways of coping eventually create a coping strategy.

You can cope with stress in positive or negative ways. A negative coping strategy might be to ignore your problems and emotions, hoping they work themselves out. But that can be bad for your health. Positive coping strategies allow you to deal with stress in healthy ways.

Managing your own emotions

The best thing you can do to combat negative moral emotions and their repercussions is understand how to deal with emotions. Take the Recognize, Understand, Manage approach when dealing with emotions at work.

Recognize

When emotions start to bubble up, don't panic. Take a deep breath and recognize the emotion for what it is. Don't react immediately. Instead try to put a label on what it is you're feeling. Determine when you became aware of the feeling and what triggered it. Don't judge yourself for how you feel.

Understand

After you've named your emotions, focus on the "why" behind them. Dig deep and try to discover their origin. Follow them down the pathway to where you are now with questions like these:

- If you feel upset, what is causing you to feel that way?
- Are your emotions coming from something within you, or something external?

If it's a familiar emotion, think about other times you've felt this emotion and how you previously responded.

- What went well?
- What didn't?
- How do you want to respond differently in this moment?

Manage

Once you've taken time to cool down and reflect, the third and final step is managing the situation. You need to figure out how you are going to respond, if at all. There are no hard and fast rules for how to respond, but here a few things to consider:

- Do you still feel the need to address the situation?
- Is it possible you overreacted?
- Are there things that need to be resolved before you can move forward?
- What will you say when you do address the situation? What might others involved say?
- What did you learn from this situation that you can apply to future situations like it?

Dealing with employee emotions: 4 tips for leaders

Regardless of how well you handle your own emotions, you can't control the emotions of others. But it is important to learn how to acknowledge them and respond appropriately. Unresolved issues can lead to decreased productivity, damaged relationships, and lowered engagement.

Follow these tips for dealing with your employees' emotions and start building a stronger emotional culture at work:

1. Allow for mistakes.

No one is perfect—plain and simple. Mistakes are inevitable and, although they don't need to be simply accepted, they also aren't grounds for a scolding. Berating or punishing employees can cause humiliation and hostility.

Calmly correcting or excusing the rare mistake is a great way to build trust. In fact, helping reduce negative emotions with transparency, open communication, and authenticity will lead to better relationships with your employees. But if mistakes continue to pile up, you should schedule time

with your employees to create a performance improvement plan. This plan will help make sure both parties are on the same page and set clear expectations for success.

2. Build a culture of trust.

Sharing emotions, especially uncomfortable ones, is one way to show vulnerability. But we can't be vulnerable if we don't trust the people that we're sharing our stories with. Everyone in your organization should feel comfortable being themselves and expressing their emotions.

It's not enough just to let employees know they can share their feelings—you must be willing to be the example too. When employees can witness a culture of honesty and compassion, they'll be able to understand and adapt to others' emotions.

3. Be present.

When managers and leaders are unavailable, employees may feel anxious and undervalued. Make time to connect with your team to instill confidence in your employees, their work, and their performance. Empowering your team members to stay connected in this way helps build positive relationships.

Set uninterrupted time apart for each employee at least once a month. And if you're really committed to the cause, we recommend meeting once a week—even for a quick check-in. These meetings offer employees an opportunity to ask questions, provide updates, raise concerns, and provide feedback. These are perfect times to celebrate successes to increase positive emotions in the workplace.

4. Listen more than you talk.

Most times, employees who experience negative emotions aren't searching for solutions. They want to express themselves and release their pent-up emotions. Listening to your employees allows them to get it all out there and makes them feel cared for and heard. It also establishes you as a trusted resource who can be depended on.

When dealing with someone else's emotions, strive to understand what they are feeling and why. Ask them how they are feeling and if they are okay. If they don't want to talk or say everything is fine, respect their wishes and don't pressure them any further. If they do want to talk it out, find a quiet space and listen.

Your employees are only human, and humans are emotional creatures. Addressing emotions is important for recognizing your employees for who they are and improving your emotional culture.



CONTEXTUAL ANALYSIS

Unit Structure

- 6.0 Objectives
 - 6.1 Introduction
 - 6.2 Abridged contextual inquiry process
 - 6.3 Data-driven vs. model- driven inquiry
 - 6.4 History
 - 6.5 Contextual Analysis
 - 6.6 Extracting Interaction Design Requirements
 - 6.7 Constructing Design- Information Models
-

6.0 OBJECTIVES

A contextual analysis is simply an analysis of a text (in whatever medium, including multi-media) that **helps us to assess that text within the context of its historical and cultural setting, but also in terms of its textuality – or the qualities that characterize the text as a text.**

6.1 INTRODUCTION

What is Contextual Research?

We can define contextual Research as, “the designer going out in the user’s natural environment observing their behaviour and the potential interactions with the interface”

Here, the designer closely maps the user journey. The designer then notes down the insights and creates a UI/UX based on the user’s requirements and expectations. Since this approach changes the dynamic of research, the data collected is immensely useful.

What are the advantages of contextual research?

This also means interacting with users to understand them better.

Interactiveness:

Contextual Research is all about stepping into the user’s shoes. This also means interacting with users to understand them better. This interaction takes us closer to the user and helps us deliver a more promising interface.

Usability:

Contextual Analysis

This research is also a measure for the usability of the app/website experience you created. Since the user is in his own environment, it's a natural setting for him/her. This familiarity helps the researcher to gain real-life insights about the usability of the design.

Unconsidered factors:

Since this research is more hands on than the others, it helps us recognise more elements than what usually meets the eye. It introduces us to several factors that we might have not considered otherwise.

Actual Actions:

Unlike any ordinary research, contextual research doesn't always make the user conscious of the research. Thus, because of the natural environment being offered, we can witness their actual journey with the app/website.



Qualitative Analysis:

It is believed that Contextual Research is one of the best forms of research. The proximity with the user helps in understanding the touch points and make the design user-friendly.

What is the Process of Contextual Research?

In practice, it is about assuring the user that it is research for the product and not the user. This will prevent pressurizing themselves and giving truthful experiences rather than made-up feedback.

Secondly, the research team should be well aware of what they exactly want to test during the research.

We can classify the flow of contextual research into three steps:

The Introduction:

This is where the researcher and user establish a relationship and discuss the purpose of the research.

Main Body:

The Main Body is the juice of the fruit. This is exactly where the research team records the user's behaviour with your product and note it down simultaneously.

Wrap-up:

This phase includes discussing the user's action with the user to clear any misconceptions. The research findings are then consolidated to apply to the product.

6.2 ABRIDGED CONTEXTUAL INQUIRY PROCESS

In a contextual inquiry, the UX researcher observes how participants perform certain tasks while having them describe what they are doing through their interaction with the product. Unlike a usability test, the user is interacting with the product in their natural habitat and context of use. This contextual environment yields richer insights and a deeper understanding behind the behavioral interaction between a user and the product. The goal is to observe the actions the users perform and understand the goals behind those actions.

When to conduct contextual inquiries

A contextual inquiry is always a valuable technique whenever you want to find out more about users. Most of the time, this is in the early stages of a project. Since UX is not linear, a contextual inquiry can also occur after a product release to help gauge the success and efficiency of your solution. For the most part, a contextual inquiry occurs at the beginning of a project when you are evaluating whether or not users would even have a need for your product. A contextual inquiry is a great way to help you validate an idea.

Another valuable reason for a contextual inquiry is that it captures how users currently solve the problem that you are aiming to fix. This will help shed light on the current pain points to help guide your design decisions moving forward. Once you understand the user's pain points (and the things that worked well), you can design a solution to help improve the existing user journey.

Two main models of contextual inquiries

A contextual inquiry is based on the interaction between the UX researcher and the participant. When taking this collaboration into consideration, there are two main modes of interaction to consider.

1. **Active inquiry** – This model occurs when the user and researcher talk through all of the tasks being performed by the participant. This provides an additional layer of information about their process. This model allows the researcher to interrupt the participant throughout the journey to ask any clarification questions they may need.
2. **Passive inquiry** – This model involves the participant performing their tasks without engagement from the researcher. The participant completes their task as if nobody is watching and the research silently observes their behavior. In this model, all questions

regarding the participant's interactions are asked at the end after the task is complete.

Contextual Analysis

The four guiding principles of contextual inquiries

There are four principles that guide an efficient contextual inquiry. These principles help keep the research focused, leading to rich, empowering data.

1. **Context** – The most unique aspect to contextual inquiries is that they take place in the context of use. The researcher is able to observe in their natural habitat and ask follow up questions either during or after the observation. Being in context also helps jog people's memories of past experiences that add additional value to this type of user research method.



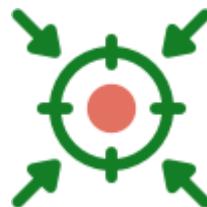
2. **Partnership** – There is a collaborative partnership that exists between the UX researcher and the participant in order to understand the user's journey. The researcher is able to work together with the participant, discussing the reasoning behind the user's behavior either during or after the observation. As mentioned, this partnership can either be in an active or passive setting.



3. **Interpretation** – The researcher will review their interpretations of the participant's actions at the end of a session. This empowers the researcher to validate or clarify their interpretations with the participant before they depart. Having a correct interpretation of the event is essential to ensure you are designing for the right reason.



4. **Focus** – An efficient user researcher will steer the interaction towards topics that are relevant to the scope of the project. The focus describes the area you want to further understand and what types of actions you need to observe to get you there. This focus can help you keep the interview centered around relevant topics without too much divergence.



The three steps to conducting a contextual inquiry

The structure of a contextual inquiry is pretty straight forward, and should be done in three parts.

1. **Introduction** – This is the stage where the researcher and the participant establish trust and communication, forming the partnership mentioned above. The researcher will introduce themselves and let the participant know the reason behind their inquiry. The researcher will typically ask permission to record the whole experience, letting them know when the recording has started and stopped. The researcher will typically explain that there are no wrong answers, aiming to make the participant comfortable before moving onwards. This is also the stage where you mention the confidentiality agreement.



2. **The process** – This is the section where the participant actually interacts with the product. Here, the researcher observes the interaction making notes of pain points as well as things that went smoothly. All of these pros and cons will aid in the direction of the design at hand. In an active Inquiry model, the researcher will ask questions as the participant interacts with the product to help gain additional insight. In a passive Inquiry model, the questions take place in the wrap up.



3. **Wrap up** – This is the final stage of a contextual inquiry, when the researcher summarizes the results of the session to the participant. Presenting these conclusions, gives the participant the opportunity to verify certain behaviors or clear up any areas of ambiguity. This empowers the participant to correct the researcher and gives them the opportunity to provide additional points that they think are important. As mentioned, in a passive inquiry, the researcher waits until the end of the interaction until they ask questions.



The benefits and limitations of a contextual inquiry

The benefits of doing a contextual inquiry are considerable, despite its limitations. Like everything, it's important to weigh your options. Keep the following in mind when deciding whether you will do a contextual inquiry.



Benefits:

- Behavior is more accurate when observed in context.
- Ability to observe nonverbal cues such as body language and facial expressions.
- Helps refine persona and scenario development prior to jumping in to the design.
- Reveals information and behaviors the participants may not be aware of.

Limitations:

- Analysis depends on the capability of the observer.
- Your mere presence influences their activity and is not 100% natural.
- Intensive use of time; Can be costly when compared to other research methods.
- Approaching participants takes a certain type of personality to do so effectively.

Six best practices to keep in mind during a contextual inquiry

Now that we are almost experts in contextual inquiries, I want to leave you with six key considerations to keep in mind in order to maximize the success of your next contextual inquiry:

1. Record when possible.
2. Speak with one participant at a time.
3. Encourage participants to think out loud.

User Experience Design 4. Notate body language and non-verbal signals.

5. Do not put participants under pressure.
6. Plan time for analyzing results.

Contextual inquiries are one of the richer forms of user research. They are a great way to validate the need for your product while gaining insight on how users currently solve the problem you are trying to fix. When used correctly, they empower the designer to make research-backed design decisions, leading to the creation of effective products that solve important pain points. Now that you understand in depth this methodology, go out and apply it during your next UX project.

6.3 DATA-DRIVEN VS. MODEL- DRIVEN INQUIRY

There are two schools of thought in AI Data-driven approach and the Model-driven approach. The data-driven approach talks about improving data quality, data governance to improve the performance of a specific problem statement. On the other hand, the model-driven approach tries to build new models and new algorithmic manipulations (or improvements) to improve performance. In an ideal scenario, both should go hand in hand but in reality, we have progressed much in terms of model-driven approach in comparison to data-driven approaches. There's still a lot to be desired in data governance, data management, data quality handling as well as the general awareness regarding that.

An example in this context can be given for the recent work done on Covid-19. While the world was reeling from the pandemic situation, a lot of AI-related work surfaced. Whether it be detecting Covid-19 from CT scan, X-ray, or other medical imaging or predicting the disease progression, or even predicting the total number of deaths. On one hand, this large amount of work across the globe has improved the knowledge base of the disease itself and in some places helped the clinical workers to work with large masses. However, at the other end, very few among the large amount of work were deemed fit for any practical implementation process like in the healthcare sector. The reason for such low practicality can be traced back to mainly data quality issues. A huge number of projects or research were done on duplicate images from other datasets. Even then, there's a distinct lack of external validation as well as demographic information on training data. Most of these projects would fail a systematic review and fail to report on the biases. Hence, the reported performance can not be translated to real-world applications.

One important aspect of Data science that needs to be remembered is that bad data can never lead to great performance however good your model is. Awareness of systematic data collection, management, and consumption is necessary for a Data Science project to have real-world applications. Only then, society can reap the benefits of the ‘marvelous AI’.

6.4 HISTORY

Many early wins in the design of computers for human use came from PARC, a Xerox research center founded in the early 1970s to explore innovations in workplace technology. PARC's work in the mid-70s produced many user interface conventions that are still used today—the graphical user interface, the mouse, and computer-generated bitmap graphics. For example, PARC's work greatly influenced the first commercially available graphical user interface: the Apple Macintosh. The term user experience probably originated in the early 1990s at Apple when cognitive psychologist Donald Norman joined the staff. Various accounts from people who were there at the time say that Norman introduced user experience to encompass what had theretofore been described as human interface research. He held the title User Experience Architect, possibly the first person to ever have UX on his business card. Norman actually started out in cognitive psychology, but his writing on the cognitive experience of products, including technological products, made him a strong voice to lead and inspire a growing field. According to Don Norman, “I invented the term because I thought Human Interface and usability were too narrow: I wanted to cover all aspects of the person’s experience with a system, including industrial design, graphics, the interface, the physical interaction, and the manual.”

6.5 CONTEXTUAL ANALYSIS

A contextual task analysis, or contextual inquiry, is a user research method that applies ethnographic observation and one-on-one interviewing to understand the task procedures that users follow to reach their goals. The researcher silently observes the user at work in his or her natural work environment and notes any tools and people that support the user as they work toward task goals.

For example, if a user is booking a vacation online, she may look up hotels online and then call the hotel front desk to learn more about the hotel. She may also review consumer ratings online to determine if the hotel offers amenities and services that meet her needs and preferences. In this example, the researcher would simply observe the user while she finds a hotel that is suitable. The researcher will take notes, and possibly use audio and/or video recording to capture key moments in the process. Following the observation, the researcher may conduct a one-on-one interview to understand the procedure from the user’s point of view. In order to analyze a task procedure in terms of the average emotional and behavior patterns experienced by all participants in a study, the researcher may conduct exercises that let users map out the task procedure on paper and indicate their emotional reactions to specific procedural steps.

Sequence Model

During contextual inquiry, users will describe several tasks that they carry out in an application or on a website in order to accomplish specific goals. Researchers may use observation, participatory exercises and structured

interviews as tools to learn more about the procedural steps that users take in order to reach their goals. The data collected can be portrayed as a sequence model to understand the sequence of steps that users take, from start to finish, as they work toward specific goals. A sequence model describes a particular task, the user's desired goal, and lists each detailed step involved in the order that the user carries them out.

Artifact Model

An artifact model depicts the physical tools used to accomplish a task. This model includes an illustration or photograph of each artifact, along with a description of how it contributes to the user's workflow. For example, a telephone may be an artifact if the user must place a call in order to complete a task.

Cultural Model

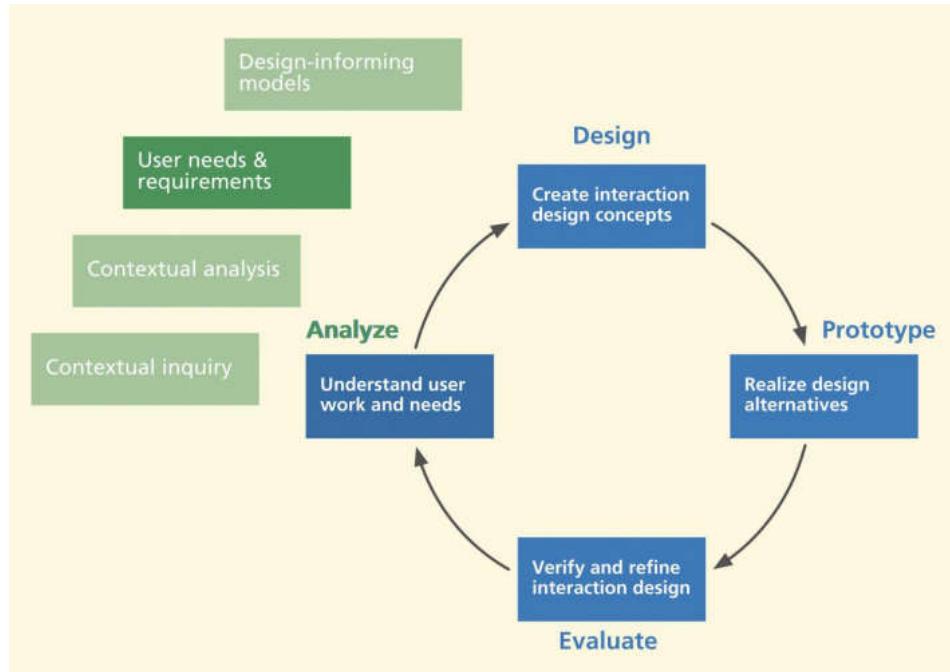
A cultural model is a diagram that shows how each group of people involved in a task work for or against one another. The diagram depicts the roles of each person and their relationships to the groups of people with whom they interact directly and indirectly. One group may follow a specific protocol to resolve a problem, and that may affect other groups involved. For example, in a call center, an employee in the Customer Service department may send a customer over to the Technical Support department. Each of these departments has their own set of rules, guidelines, and etiquette, which all affect the customer's experience. Even the process of transferring a customer from one department to another in order to solve their problem says something about the culture of the company itself. When these cultural relationships are mapped out, they can be examined to improve efficiency and user satisfaction.

Physical Model

A physical model portrays the physical characteristics of a work environment or communication network. A physical model is a sort of blueprint that depicts the workflow in terms of where certain artifacts are located and how the user and other equipment, such as computer workstations, servers, printers, etc., work together to accomplish a task. For example, a user may access a website from his home, and the server storing the website he is viewing may be located in a city far from his home. The content of the website may be updated at the office of a company or web developer in yet another city. A physical model of this scenario would map out what each person or tool does at each location and how they relate to one another. This document may be used to ensure that the workflow and processes involved run efficiently and effectively.

6.6 EXTRACTING INTERACTION DESIGN REQUIREMENTS

Contextual Analysis



UX Requirements means interaction design requirements -

What is required to support user work activity needs

- Work activity notes are not requirements
- Requirements bridge contextual inquiry and analysis to design
- What to look for?
 - Functionality of course
 - Usability goals
 - Emotional impact – “fun”, “boring”, ...

UX Requirements Extraction: How?

- Walk the WAAD one note at a time to deduce needs and UX requirements
- Filter terminology to achieve consistency (e.g. alarm, alert)
- What user needs are implied by the work activity note? Translate each user need into one or more interactive design requirements
- **Switch from inductive to deductive reasoning**
- What UX requirement is “deduced” from a work activity note in a WAAD?

- Consolidate notes to condense ideas
- Extrapolate notes to broaden

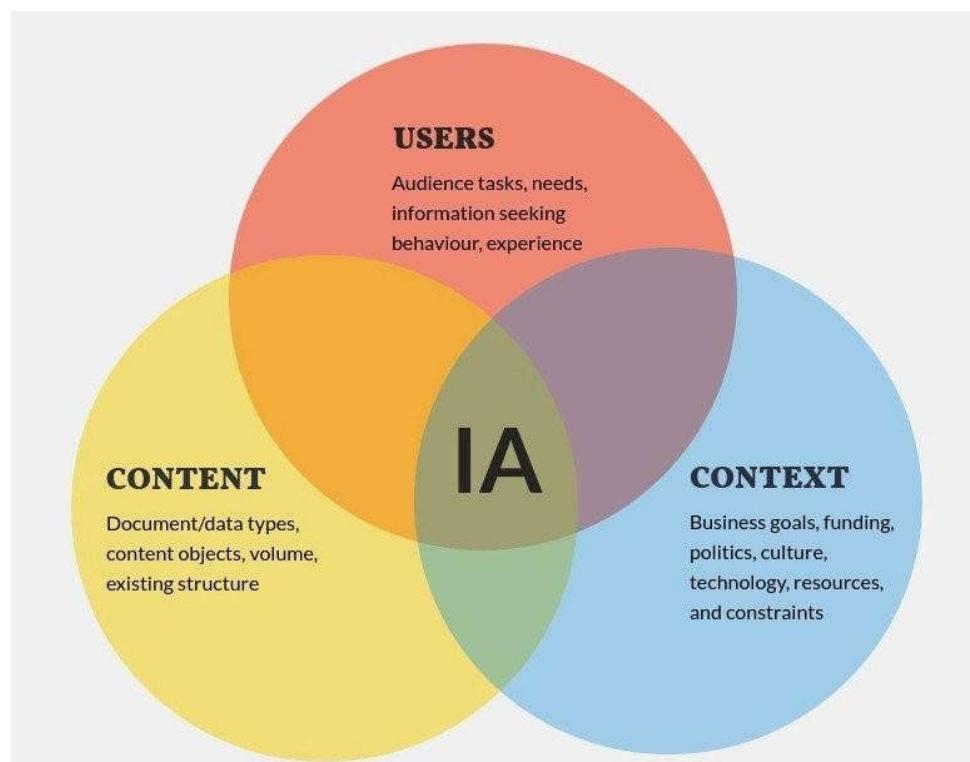
6.7 CONSTRUCTING DESIGN- INFORMATION MODELS

What is information architecture?

When you arrive at a new place, you need something to guide you. If you're spending a weekend in an unfamiliar city, you have to consult a map. Then you use street signs and addresses on buildings to reach your destination. It's the same with any unfamiliar location: You don't want to waste time and get lost, so you need guidance.

Information architecture (IA) is the science of structuring content as it is applied to news websites or blogs, online stores, booking apps, downloadable software, etc. The goal of information architecture is to classify the content in a clear and understandable way and arrange it according to relations between the content pieces, allowing users to find what they need with less effort. Not applied solely when creating a product from scratch, IA is used in redesign.

Information architecture is a part of interaction design that considers content, context, and users. This means that user needs, business goals, and different types of content must be taken into account while structuring a product's information.



Main components of information architecture, source: Lucia Wang

Usually, information architecture design falls under the care of UX and UI designers or an information architect. To avoid confusion, let's briefly go over the differences between these similar terms.

Information architecture vs UX vs UI

IA, UX, and UI are closely related and sit under the umbrella of product design concept, but they are not the same. Check the definitions.

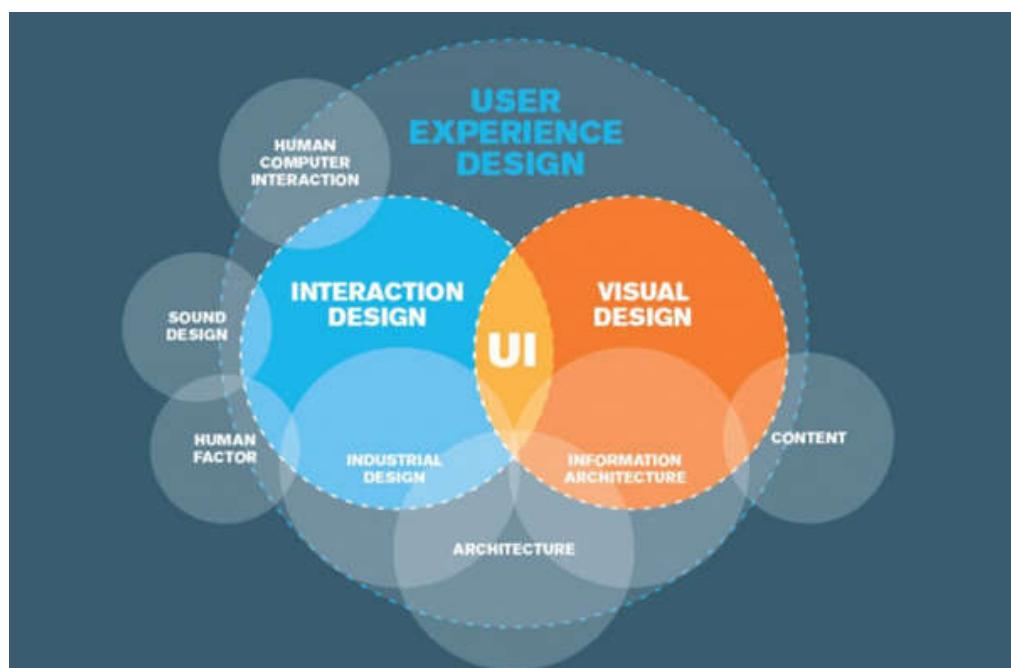
UX or user experience is what users feel when they interact with the product. So, the aim of UX design is to make a product practical, useful, attractive, and so on — in other words, to create a positive experience from using it.

UI or user interface is an important part of UX since it deals with the visual aspect of the product and the interactivity behind it. So, UI design directly impacts UX since appealing graphics and easy, intuitive interaction are integral parts of a smooth user experience.

IA or information architecture refers to organizing and labeling content to make the product usable and understandable, while enhancing user experience. So, in a way, IA design is a foundation of UX.

To make it clearer, here's an analogy. You can compare IA with the skeleton of the product that supports and holds all parts in the right place. Then, UI is the skin or appearance that other people see as they interact with the product. And UX is the emotion that is created as a result of such interaction.

Here's an excellent visualization of the interrelation of these terms.



How IA, UX, and UI are related, image credit: Scorch

Peter Morville, the co-author of *Information Architecture for the World Wide Web*, explains the role of an information architect as a person who bridges users and content by designing search and navigation, embodying the abstract ideas into prototypes, units, and disciplines to turn the concepts into something understandable. An information architect who works along with a UX designer can concentrate solely on information architecture design, while a UX designer devotes more time to research. In this case, an architect creates a number of deliverables, which we will describe in the section on the steps of IA development.

Regardless of the title, a person who works on information architecture must start with the rules that help to achieve the goal of IA. In the next part of the article we will look closer at the activities of IA design and its deliverables. To create a strong information architecture, you must start with thorough research, so it all begins with learning your users' needs and behaviors.



ARCHITECTURE AND INTERACTION DESIGN OF UX

Unit Structure

- 7.0 Objectives
 - 7.1 Introduction
 - 7.2 Information
 - 7.3 Architecture and Interaction design
 - 7.4 Prototyping design
 - 7.5 Design paradigms
 - 7.6 Design thinking
 - 7.7 Design perspectives
-

7.0 OBJECTIVES

The goal of UX design in business is to “**improve customer satisfaction and loyalty through the utility, ease of use, and pleasure provided in the interaction with a product.**”

7.1 INTRODUCTION

User experience (UX) design is the process design teams use to create 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.

What UX Designers do goes Beyond UI Design

“User Experience Design” is often used interchangeably with terms such as “User Interface Design” and “Usability”. However, while usability and user interface (UI) design are important aspects of UX design, they are subsets of it – UX design covers a vast array of other areas, too. A UX designer is concerned with the *entire* process of acquiring and integrating a product, including aspects of branding, design, usability and function. It is a story that begins before the device is even in the user’s hands.

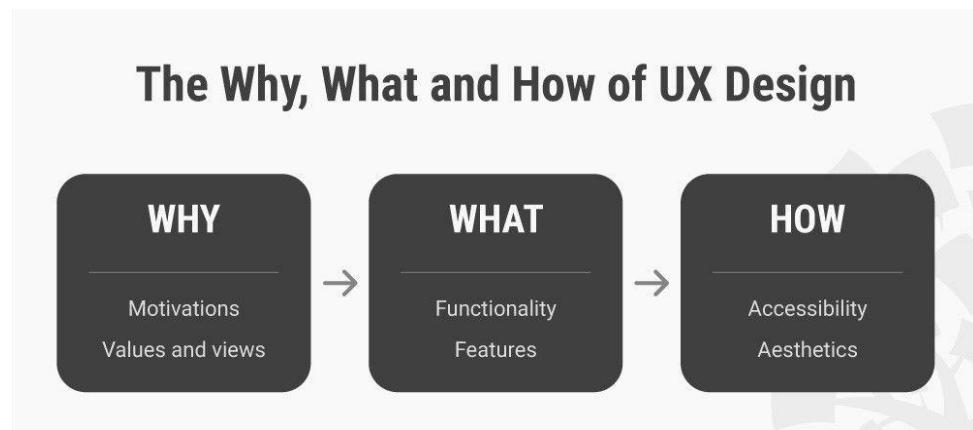
“No product is an island. A product is more than the product. It is a cohesive, integrated set of experiences. Think through all of the stages of a product or service – from initial intentions through final reflections, from first usage to help, service, and maintenance. Make them all work together seamlessly.”

— Don Norman, inventor of the term “User Experience”

Products that provide great user experience (e.g., the iPhone) are thus designed with not only the product’s consumption or use in mind but also the entire process of acquiring, owning and even troubleshooting it. Similarly, UX designers don’t just focus on creating products that are usable; we concentrate on other aspects of the user experience, such as pleasure, efficiency and fun, too. Consequently, there is no single definition of a good user experience. Instead, a good user experience is one that meets a particular user’s needs in the specific context where he or she uses the product.

UX Designers consider the Why, What and How of Product Use

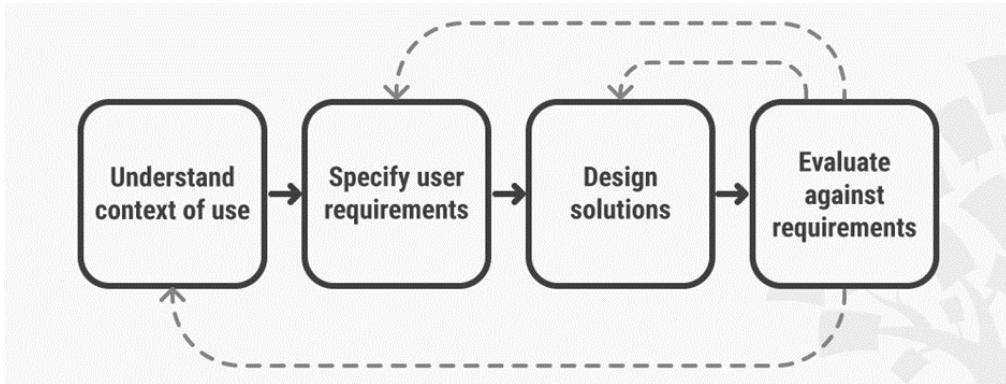
As a UX designer, you should consider the Why, What and How of product use. The Why involves the users’ motivations for adopting a product, whether they relate to a task they wish to perform with it or to values and views which users associate with the ownership and use of the product. The What addresses the things people can do with a product—its functionality. Finally, the How relates to the design of functionality in an accessible and aesthetically pleasant way. UX designers start with the Why before determining the What and then, finally, the How in order to create products that users can form meaningful experiences with. In software designs, you will need to ensure the product’s “substance” comes through an existing device and offers a seamless, fluid experience.



UX Design is User-Centered

Since UX design encompasses the entire user journey, it’s a multidisciplinary field – UX designers come from a variety of backgrounds such as visual design, programming, psychology and interaction design. To design for human users also means you have to work with a heightened scope regarding accessibility and accommodating many potential users’ physical limitations, such as reading small text. A UX designer’s typical tasks vary, but often include user research, creating personas, designing wireframes and interactive prototypes as well as testing designs. These tasks can vary greatly from one organization to the next, but they always demand designers to be the users’ advocate and

keep the users' needs at the center of *all* design and development efforts. That's also why most UX designers work in some form of user-centered work process, and keep channeling their best-informed efforts until they address all of the relevant issues and user needs optimally.



User-centered design is an iterative process where you take an understanding of the users and their context as a starting point for all design and development.

7.2 INFORMATION

In recent years, the term '**user experience design**' has become synonymous with technology and software, but this wasn't always the case.

Despite how we use the terminology now, user experience was a term that simply applied to how a person feels about using a system.

The term 'user experience design' was first coined by **Don Norman** in 1995 while he was the vice president of the Advanced Technology Group at Apple. He said:

"I invented the term because I thought human interface and usability were too narrow. I wanted to cover all aspects of the person's experience with the system including industrial design, graphics, the interface, the physical interaction, and the manual."

Norman also authored the book 'The Design of Everyday Things' which pioneered the prioritization of usability and function over aesthetics and remains highly influential in design circles today.

As interest in the field has grown, "UX" has become more of an umbrella term for a number of different fields, such as **User Research**, **Information Architecture**, **Usability Engineering**, **Service Design**, and so on.

7.3 ARCHITECTURE AND INTERACTION DESIGN

The interaction design (IxD) process is what designers use to create solutions centered on users' needs, aims and behavior when interacting with products. The IxD process involves 5 stages: discovering what users

User Experience Design need/want, analyzing that, designing a potential solution, prototyping it and implementing and deploying it.

The 5 Stages of the Interaction Design Process

With the IxD process, you can build highly intuitive, recognizable interfaces that provide seamless experiences for users and prove your brand thoroughly understands them, their contexts and the goals they seek to achieve.

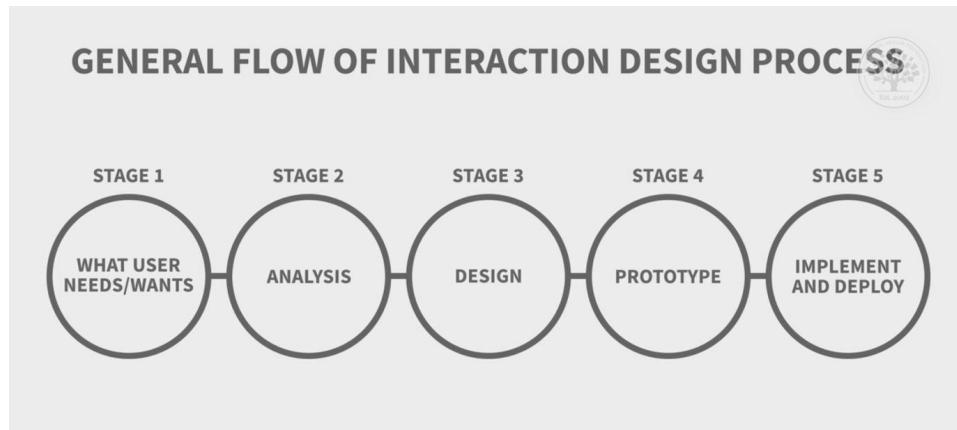
Here are the five stages that the IxD process typically involves:

1. **Find the users' needs/wants**—It's easy to assume you know what users want/need and their relevant contexts. Discover their *real* requirements:
 - . **Observe people.**
 - a. **Interview people.**
 - b. **Examine existing solutions**—while remembering it's hard to envisage future needs, technologies, etc.
2. **Do analysis to sort and order your findings so they make sense.** This may be through a:
 - . Narrative/story of how someone uses a system.
 - a. Task analysis, breaking down a user's steps/sub-steps.
2. **Design a potential solution according to design guidelines and fundamental design principles** (e.g., giving appropriate feedback for users' actions). Use the best techniques to match how users will interact with it in terms of, for example, navigation.
3. **Start prototyping**—Give users an idea of what the product will look like and let them test it, and/or give it to experts to evaluate its effectiveness using heuristics.
4. **Implement and deploy what you have built.**

The IxD process is iterative—nobody designs anything right the first time, especially regarding more innovative solutions. It may indeed take many iterations before you pinpoint the ideal version of a solution. So, you (and your design team) should continue testing and adapting appropriate changes around an ever-clearer understanding of your users' needs. For example, you could gather user feedback and monitor support chats to find areas for improvement.

It's important to understand the interaction design process is a *general* idea of how you can start from your users' needs and progress towards a fitting solution. Similar design processes exist. Design thinking is one of the more notable of these, where you work to gain and leverage vital insights to fine-tune optimal features. Only when you know your users

and empathize with them can you appreciate their real-world needs, desires and pain points.



7.4 PROTOTYPING DESIGN

A *prototype* is a primitive representation or version of a product that a design team or front-end-development team typically creates during the design process. The goal of a prototype is to test the flow of a design solution and gather feedback on it—from both internal and external parties—before constructing the final product. The state of a prototype is fluid as the team revises the design iteratively based on user feedback.

Prototyping is an integral part of the design process for two key reasons:

1. Visualization—Prototypes help UX designers show stakeholders how the final product would look and function.
2. Feedback—Prototypes generate feedback from team members as well as test groups of users. Potential customers can interact with a near-final product and highlight areas that are less than user friendly. The design team can then iterate the design before the product team rolls out the final product, saving the company both time and money.

Advantages of Prototyping

1. Validation of research findings

Research conducted during the early stages of a project does not tell us everything about the optimal solution. A lot of positive feedback or glitches are discovered only once we test. By prototyping and then testing those prototypes, we can uncover far richer insights and validate findings from initial research.

2. Applications

Prototyping can be used to test a different kind of ideas, concepts in different forms as well as applicable at different stages of design and redesign process.

3. Issues and error identification

Prototyping allows identifying issues as well as errors or biases that may have been introduced into research during the early stages of solution formulation.

4. End-User engagement

Prototyping helps engaging with potential customers and getting first hand feedback on artifacts. This also gives deeper insights and a better idea of the value that can be captured from a proposed solution.

Disadvantages of Prototyping

1. Added time and costs

Prototyping is not as inexpensive as it sounds. Even though, prototyping helps making informed decisions about the design direction for a product or service, it is still an additional cost as recruiting users, testing, making alterations and testing again can take up a lot of time as well as investments not initially anticipated.

2. End-user recruitment

The kind of testing involved and the quality of feedback collected depends a great deal on the end-users that are recruited to test a solution. If for some reason, the recruitment isn't accurate or doesn't involve a large set of users then important insights could get missed.

7.5 DESIGN PARADIGMS

UX Paradigms

The Audi brand is not only defined by its visual appearance. When it comes to interactive channels the question also has to be asked:\\ what defines interaction with the brand across all applications and devices? We set ourselves the goal of making interaction as simple and as positive as possible for our customers so that the experience is associated with the brand, thereby supporting lasting loyalty. On the path to achieving this goal, the UX paradigms provide a shared basis for developing new ideas and striving for constant improvement.

1. *Personalization*

We personalize our service so as to provide each customer with an individual user experience. Habits, needs, individual mobility, personal devices, security and privacy: all these differ from one user to the next. So we enable users to change or supplement content according to their individual preferences.

2. *Context*

Users can choose individually from a range of different interests, positions and objectives. They may also take on different roles such as potential customer, buyer or owner. To create a positive experience, it is important to understand and incorporate the needs and aims of the user in the design.

3. *Consistency*

We use a clear visual language to users on all channels and offer recurring solutions. Entire sets of content, hierarchies and functions are structured identically across all devices. We give functions the same name and apply animations in the same way. In this way, we make it easier for users to perform complex tasks and we create a uniform overall experience.

4. *Clarity*

Clear terms in unambiguous language help establish clarity when performing a task. We focus on the essentials and guide the user through particular highlights and animations by means of a clear-cut hierarchy. We provide clear feedback to user input and show where a piece of information or state has come from, how up-to-date it is, what it means and how it can be changed.

5. *Reliability*

Transparency and data protection are very important to the user. Since more and more services are interconnected across different systems and devices, the user has to be able to rely on every element in the ecosystem. By clearly communicating the system states we are able to give the user the opportunity to view all the relevant information. In the event of problems, we notify the user immediately and provide support in solving the problem, as well as treating user information as confidential.

6. *Details*

Details are not optional: they are a fundamental component of our visual appearance. We apply details such as animations selectively so as to create a high-quality, thrilling and hallmark overall user experience.

7.6 DESIGN THINKING

Design thinking is a non-linear, iterative process that teams use to understand users, challenge assumptions, redefine problems and create innovative solutions to prototype and test. Involving five phases—Empathize, Define, Ideate, Prototype and Test—it is most useful to tackle problems that are ill-defined or unknown.

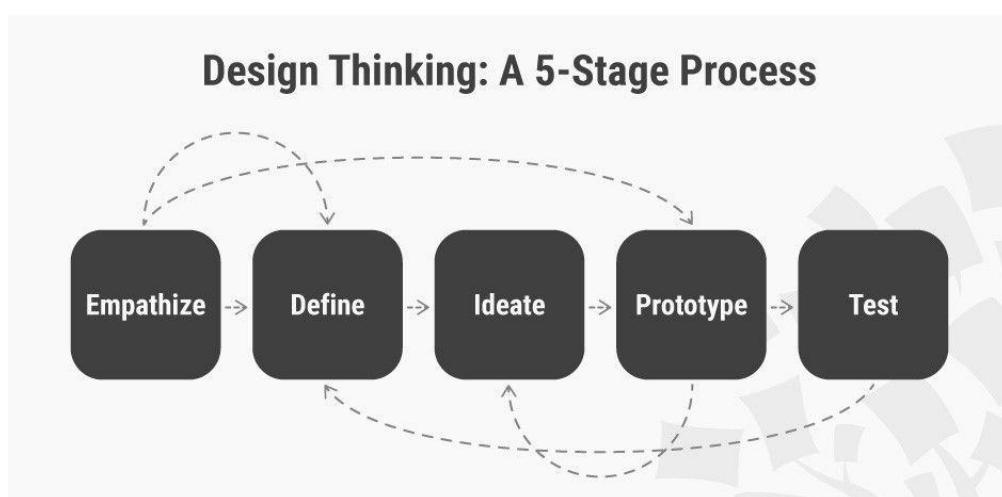
Why Is Design Thinking so Important?

In user experience (UX) design, it's crucial to develop and refine skills to understand and address rapid changes in users' environments and behaviors. The world has become increasingly interconnected and complex since cognitive scientist and Nobel Prize laureate Herbert A. Simon first mentioned design thinking in his 1969 book, *The Sciences of the Artificial*, and then contributed many ideas to its principles. Professionals from a variety of fields, including architecture and engineering, subsequently advanced this highly creative process to address human needs in the modern age. Twenty-first-century organizations from a wide range of industries find design thinking a valuable means to problem-solve for the users of their products and services. **Design teams use design thinking to tackle ill-defined/unknown problems (aka wicked problems) because they can reframe these in human-centric ways and focus on what's most important for users.** Of all design processes, design thinking is almost certainly the best for "thinking outside the box". With it, teams can do better UX research, prototyping and usability testing to uncover new ways to meet users' needs.

Design thinking's value as a world-improving, driving force in business (global heavyweights such as Google, Apple and Airbnb have wielded it to notable effect) matches its status as a popular subject at leading international universities. **With design thinking, teams have the freedom to generate ground-breaking solutions.** Using it, your team can get behind hard-to-access insights and apply a collection of hands-on methods to help find innovative answers.

The Five Stages of Design Thinking

The Hasso Plattner Institute of Design at Stanford (aka the d.school) describes design thinking as a five-stage process. Note: These stages are *not* always sequential, and teams often run them in parallel, out of order and repeat them in an iterative fashion.



Stage 1: Empathize—Research Your Users' Needs

Architecture and Interaction
design of UX

Here, you should gain an empathetic understanding of the problem you're trying to solve, typically through user research. Empathy is crucial to a human-centered design process such as design thinking because it allows you to set aside your own assumptions about the world and gain real insight into users and their needs.

Stage 2: Define—State Your Users' Needs and Problems

It's time to accumulate the information gathered during the Empathize stage. You then analyze your observations and synthesize them to define the core problems you and your team have identified. These definitions are called problem statements. You can create personas to help keep your efforts human-centered before proceeding to ideation.

Stage 3: Ideate—Challenge Assumptions and Create Ideas

Now, you're ready to generate ideas. The solid background of knowledge from the first two phases means you can start to “think outside the box”, look for alternative ways to view the problem and identify innovative solutions to the problem statement you've created. Brainstorming is particularly useful here..

Stage 4: Prototype—Start to Create Solutions

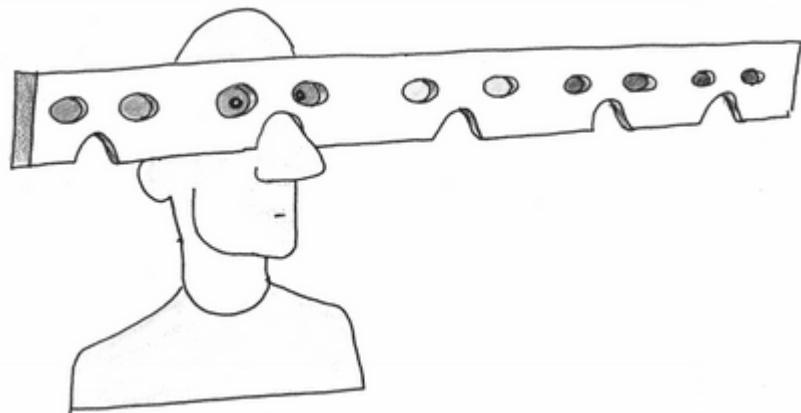
This is an experimental phase. The aim is to identify the best possible solution for each problem found. Your team should produce some inexpensive, scaled-down versions of the product (or specific features found within the product) to investigate the ideas you've generated. This could involve simply paper prototyping.

Stage 5: Test—Try Your Solutions Out

Evaluators rigorously test the prototypes. Although this is the final phase, design thinking is iterative: **Teams often use the results to redefine one or more further problems**. So, you can return to previous stages to make further iterations, alterations and refinements – to find or rule out alternative solutions.

7.7 DESIGN PERSPECTIVES

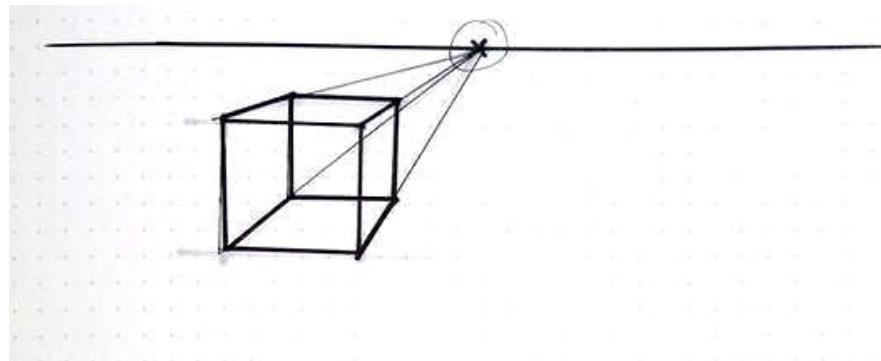
A design is a plan or specification for the construction of an object or system or for the implementation of an activity or process, or the result of that plan or specification in the form of a prototype, product or process. The design usually has to satisfy certain goals and constraints, may consider aesthetic, functional, economic, or socio-political considerations, and is expected to interact with a certain environment.



The process of designing a space is known as perspective. It all depends on how we see space and perceive it. It is one of the most fundamental elements of design that we need to understand to create realistic and visually pleasing designs.

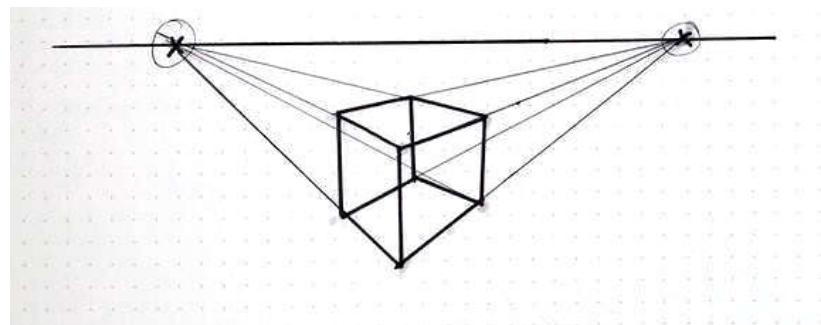
Types of perspective:

1. One-point perspective



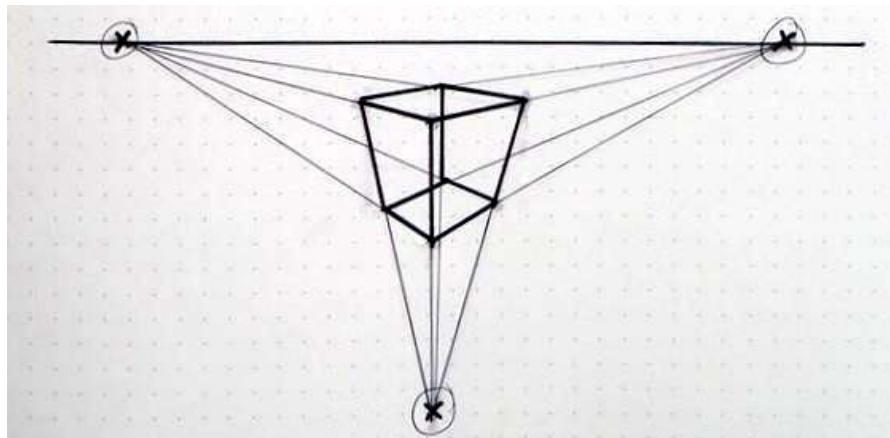
A one-point perspective means that the drawing has a single vanishing point, usually directly opposite the viewer's eye and usually on the horizon line. All lines parallel with the viewer's line of sight recede to the horizon towards this vanishing point.

2. Two-point perspective



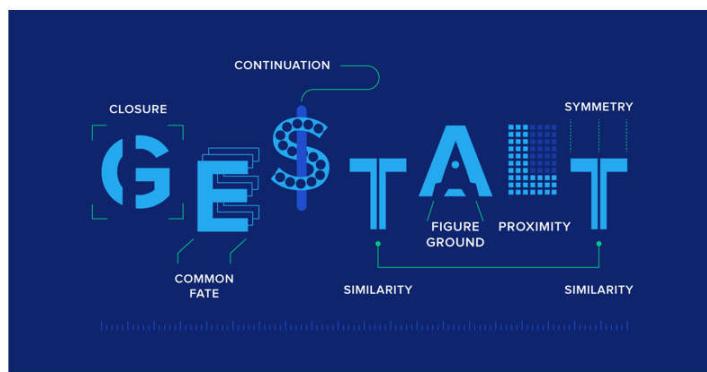
Two-point perspective occurs when you can see two vanishing points from your point of view. Two-point perspective drawings are often used in architectural drawings and interior designs; they can be used for drawings of both interiors and exteriors.

3. Three-point perspective

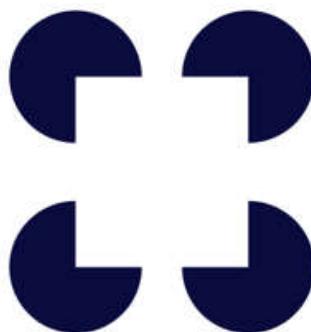


Three-point perspective has three vanishing points. Two vanishing points somewhere on the horizon; however, unlike two-point perspective, there also exists a vanishing point above or below the horizon line that the vertical lines disappear to.

Gestalt principles of design:

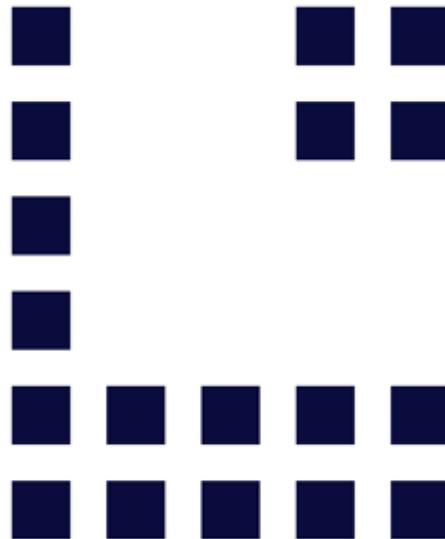


1. Closure



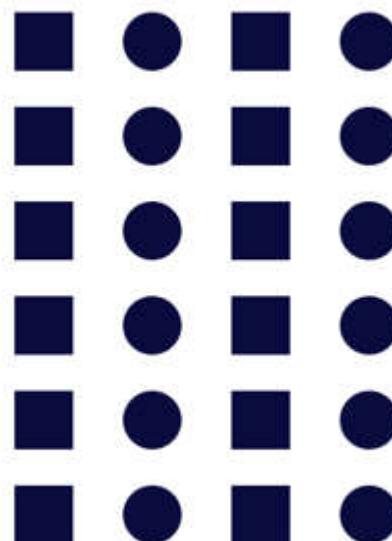
This principle states when we see any complex structure or any pattern, our brain tries to look for the recognizable pattern.

2. Proximity



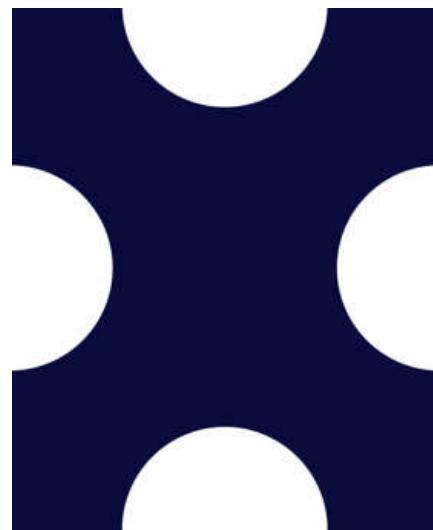
This principle states that objects that are close together appear to be more related than objects are in distance, don't share the same hierarchy, or different in color.

3. Similarity



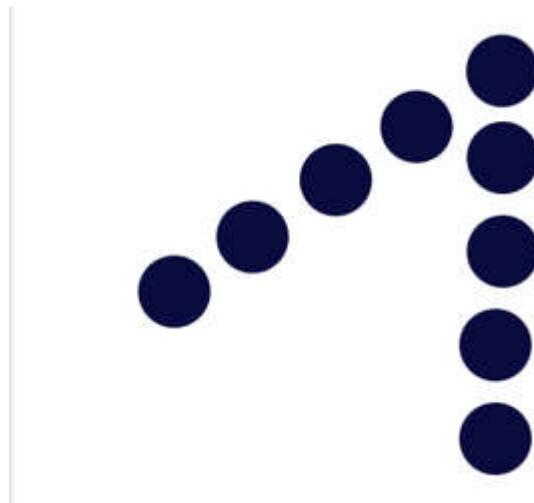
This principle states objects that share similar visual properties appear to be similar regardless of their proximity to each other.

4. Figure-ground

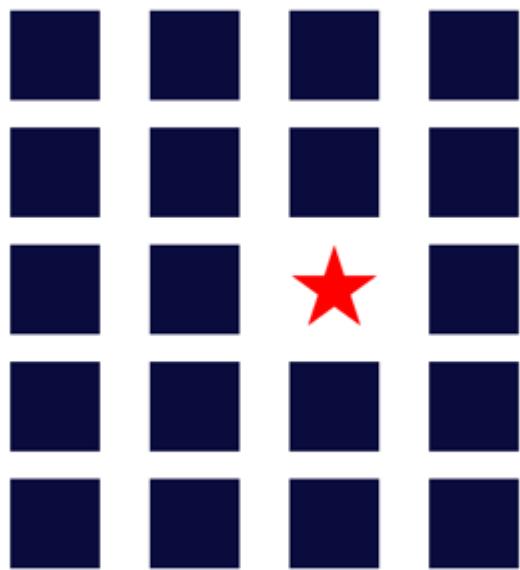


This principle states how our brain interprets between a design's foreground and background. Usually, we interpret the smaller area as foreground and the larger area as background.

5. Common fate

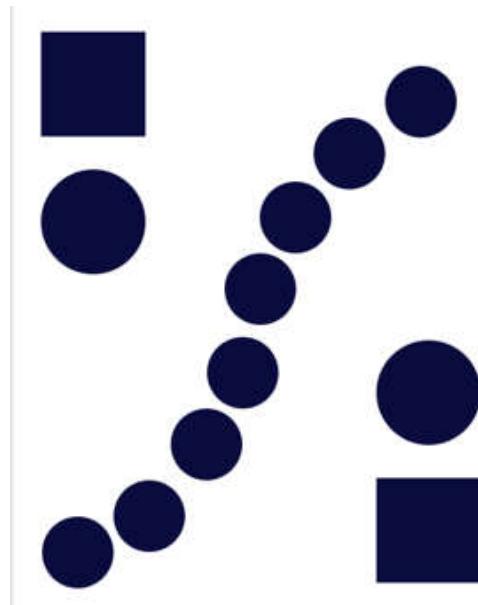


This principle states that people group together objects that point towards or moving in the same direction. They could be a bunch of different objects but because they move like a grouped object, or point towards the same direction we consider them a single phenomenon.



This principle states that in a group of objects or a composition, whatever stands out visually or captures the viewer's attention first that is Focal Point.

7. Continuity



This principle states that the human eye will follow the similarities or a line that goes from one end to another. Continuity is a collection of objects that form a shape of a line or curve related to each other than other elements on the frame.



8

THE UX DESIGN PROCESS

Unit Structure

- 8.0 Objectives
 - 8.1 Introduction
 - 8.2 User personas
 - 8.3 Ideation
 - 8.4 Sketching
 - 8.5 More about phenomenology
 - 8.6 Mental models and Conceptual design
 - 8.7 Wireframe
 - 8.8 Prototyping
-

8.0 OBJECTIVES

- To know more about UX designing process.
 - To understand about User personas, Ideation and Sketching.
 - To know more phenomenology.
-

8.1 INTRODUCTION

User Experience design is design that is user centered. The goal is to design artifacts that allow the users to meet their needs in the most effective efficient and satisfying manner. Its goal is to introduce the novice to a cycle of discovery and evaluation and a set of techniques that meet the user's needs.

Four step user interface design cycle is a set of techniques to gather information about:

Step1: Requirements Gathering

We start with requirement gathering phase that refer to understanding the user and what her goals are and what are the current practices. In this phase the designer aims to understand how the user is currently completing the task. Here we have the designers as detective, trying to understand the who, what, when, where of the user's goals. We accomplish this via set of techniques where we sometimes observe the users and other times we engage with them directly. We ask them questions or we might observe their current practices. The information we

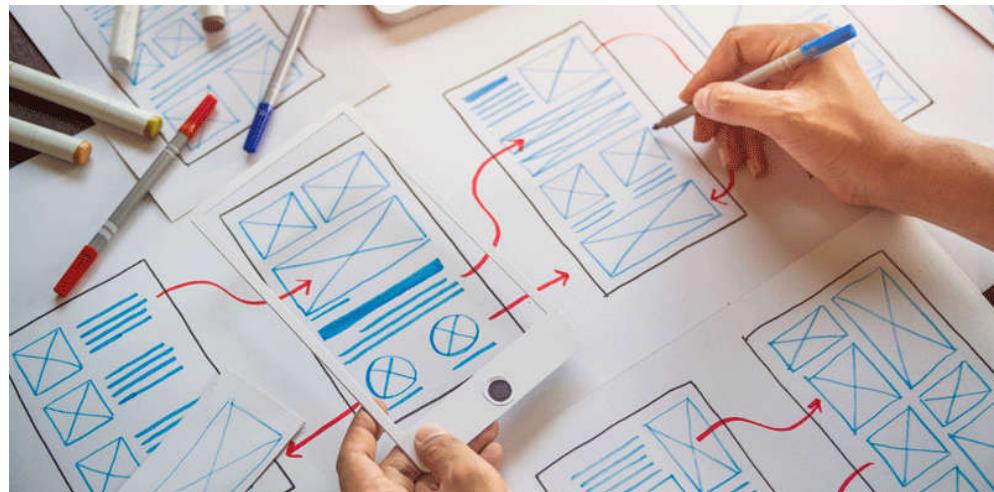
collect in this phase will allow us to understand the problem space. I like the notion of designer as detective because sometimes what the user needs is a mystery, even to the user himself, and the goal of the designer is to deduce what the best interface is to accomplish the task.

This step can also be thought of as understanding the “problem space” - what is hindering the completion of the task how can the task or process be improved. A whole host of techniques are presented that allow the designer to collect data about the user, her goals and current practices

Step 2: Design Alternatives

In the second phase, designing alternatives, we are able to develop novel interfaces to successfully complete the task. Once you understand the users, their goals, and their current practices (i.e., the problem space) you are able to take this data and develop various design options that will improve the user experience.

Step 3: Prototyping



Techniques for modelling the novel designs before a final version is produced in the third phase, the designer takes the best designs from phase two and actually builds models or prototypes that the user will be able to engage with. These models meet core aspects of the task.

Step 4: Evaluation

In the fourth phase, we take one of the designs from phase three and we actually test the system’s usability and usefulness with either users or experts who can provide us with feedback about how to improve the system.



Personas are fictional characters, which you create based upon your research to represent the different user types that might use your service, product, site, or brand in a similar way. Creating personas will help you understand your users' needs, experiences, behaviors and goals. Creating personas can help you step out of yourself. It can help you recognize that different people have different needs and expectations, and it can also help you identify with the user you're designing for. Personas make the design task at hand less complex, they guide your ideation processes, and they can help you to achieve the goal of creating a good user experience for your target user group.

As opposed to designing products, services, and solutions based upon the preferences of the design team, it has become standard practice within many human-centered design disciplines to collate research and personify specific trends and patterns in the data as personas. Hence, personas do not describe real people, but you compose your personas based on actual data collected from multiple individuals. Personas add the human touch to what would largely remain cold facts in your research. Creating persona profiles of typical or atypical (extreme) users will help you understand patterns in your research, which synthesizes the types of people you seek to design for. Personas are also known as model characters or composite characters.

Personas provide meaningful archetypes which you can use to assess your design development against. Constructing personas will help you ask the right questions and answer those questions in line with the users you are designing for. For example, “How would Peter, Joe, and Jessica experience, react, and behave in relation to feature X or change Y within the given context?” and “What do Peter, Joe, and Jessica think, feel, do and say?” and “What are their underlying needs we are trying to fulfill?”

8.3 IDEATION

Ideation is a creative process where designers generate ideas in sessions (e.g., brainstorming, worst possible idea). It is the third stage in the Design Thinking process. Participants gather with open minds to produce as many ideas as they can to address a problem statement in a facilitated, judgment-free environment.

It's challenging to gain the perspective to find design solutions. To have productive ideation sessions, you'll need a dedicated environment for standing back to seek and see every angle. First, though, your team must define the *right* problem to address. Ideation, or "Ideate", is the third step in the Design Thinking process – after "Empathize" (gaining user insights from research/observation) and "Define" (finding links/patterns within those insights to create a meaningful and workable problem statement or point of view).

Before starting to look for ideas, your team needs a *clearly* defined problem to tackle – a focused problem statement or point of view (POV) to inspire and guide everyone. "How might we...?" questions—e.g., "How might we design an app finding cheap hotels in safe neighborhoods?"—help in reframing issues and prompting effective collaboration towards potential solutions. To bring people together to conjure ideas and bypass established frontiers, you need a **skilled facilitator** and a **creative environment**, including a **prepared space**, featuring posters of personas, relevant information, etc. Your team also requires **rules** – e.g., a 2-hour time limit, quantity-over-quality focus, ban on distractions such as phones, and "There are *no* bad ideas" mindset. By being **bold** and **curious**, participants can **challenge** commonly held beliefs and **explore** possibilities past these obstacles. Team members should take each other's ideas and build on them, find ways to link concepts, recognize patterns and flip seemingly impossible notions over to reveal new insights.

8.4 SKETCHING

Sketching is a distinctive form of drawing which designers use to propose, explore, refine and communicate ideas. As a UX designer, you too can use sketching as your first line of attack to crack a design problem.

Sketches are easy, fast, and cheap to create, iterate, and if needed, even discard without much effort. Unlike written or verbal communication, sketches sidestep rules of grammar and help clearly communicate ideas, all but eliminating misunderstandings. And the best part, they are a joy to create and document!

5 Uses for Sketching in Design

There are multiple uses for sketching in the design process. Below is a review of five categories of uses with examples and links.

1. Rapid Concept Development

Sketching is an excellent way to quickly explore concepts. You can sketch for one or two hours and work out multiple possible solutions to the design problem at hand. This is an essential step in the design process. It will save you time to work through concepts on paper before going to the computer. While it is possible to build sketches on the computer, it's not as fast as sketching multiple concepts on paper.

52



53



54



The UX design process

55



56



57



58



59



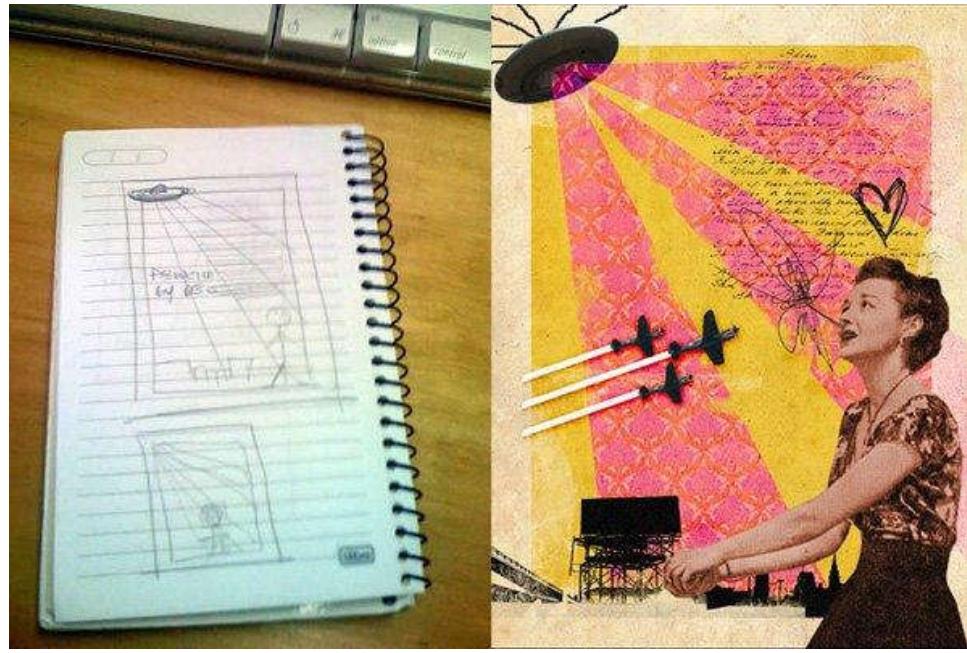
Product designers spend a lot of time sketching. If you're going to design the next sport shoe, piece of furniture, or bike, the idea doesn't start in a computer, it starts on paper.



Thumbnail sketches

2. Basic Composition or Layout

Sketches are a quick way to create the basic composition of your illustration. They are also used in Web site design and graphic design to quickly evaluate layout choices. You can make a series of thumbnail sketches, or they can be larger. As long as your sketches are good enough that they capture the necessary elements, drawing skill is unnecessary.



Creating A Cool Vintage Collage Design In Photoshop

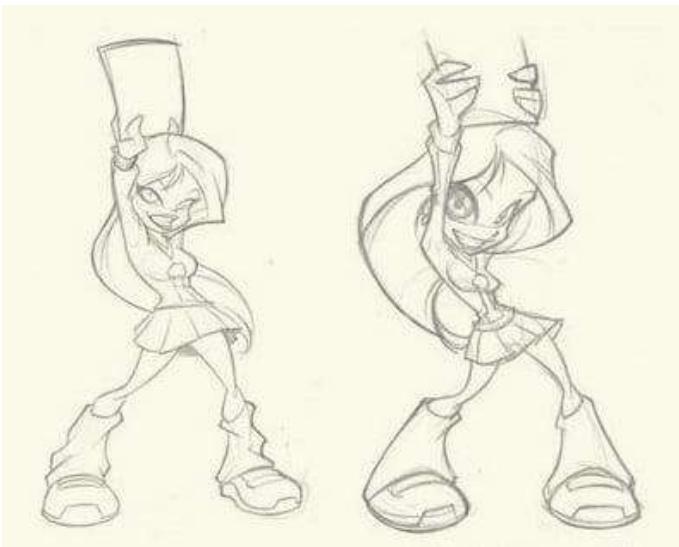
Web Design from Scratch is a well-known Web site that offers practical advice on building Web sites. In the article **The Complete No-Nonsense Guide to Designing Websites**, the author has this to say about pencil sketching layouts: "The quick pencil sketch just helps me quickly record the likeness of what I've visualized in my head. Then I don't forget and can make it up quickly in Photoshop. I find this way of working a lot more efficient than starting off in Photoshop." As you can see below, drawing skill isn't necessary to capture layout composition either. The left side below is the sketch, and the right side is the final design.



3. Client Communication and Approval

Showing sketched thumbnails or compositions to clients, will potentially save you an enormous amount of time. The more detailed the project will be the earlier you want client approval. If you're going to spend hours on an illustration, you want to make sure the client is in agreement with your choice of design before moving forward. Getting thumbnail approvals from clients is a common part of the illustration process. It is also common on large logo design projects and other projects as well.

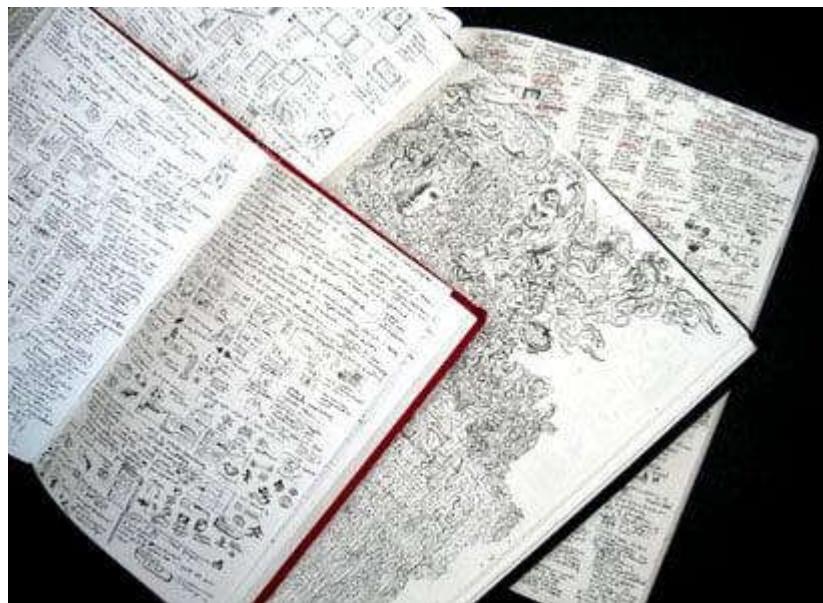
The client approves artwork before it goes to the next stage of inking and coloring. This saves time by solidifying an idea before going on to more advanced stages in the process. The example below is a concept worked out based on initial client communication. This sketch is then sent to the customer for approval or for change requests. Once the sketch is finalized, the design moved to the next stage of inking the line work and then coloring the character.



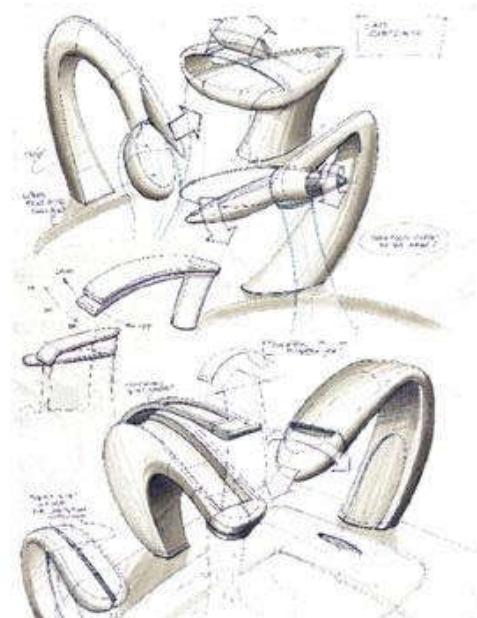
4. Visual Exploration

Sketching can be used as a journaling activity to record and explore your interests. It can also be used to explore multiple options you could take in a particular design.

Sherrie Thai has a portfolio over at Coroflot. She has a section there dedicated to **Sketches**. These sketches show her visual explorations in multiple fields of design. In the sketch area of her portfolio, she visually explores topics such as patterning, identities, and tattoo styles.



The product design book **Design Sketching** explains the entire process of sketching for product design. It offers tutorials, explanations, and examples. The example below from the book shows how a designer might investigate a problem and explore potential solutions.



5. Refining Visual Solutions

The UX design process

The process of creating a design or illustration at later stages involves refinement. The overall concept and direction of the piece may be working great, but one element isn't. Often, this can be tightened up and corrected in further rounds of sketching. Of course, at some point a digital artist moves to the computer. The process of sketching then moves into digital drafts.

8.5 MORE ABOUT PHENOMENOLOGY

Phenomenology is a philosophical discipline that studies the structures of conscious experience. It emphasizes the importance of descriptive first-person accounts of subjective experiences as a means to understand the world. Keeping in mind a phenomenological approach to UX design can help designers:

- 1) Define what **empathy** looks like in their own design process.
- 2) Ask critical questions about the **societal impact of design**.

The Project of Phenomenology

It is a commonly held sentiment that *true* knowledge about the world comes from objective facts that can be measured by tools and instruments. Phenomenology argues, that although this type of knowledge is incredibly important, it is not our only access to a true understanding of the world. Knowing the number of rain droplets that will fall from the sky would not get us any closer to understanding the phenomena of rain than being outside during a rainstorm would. People first encounter the world through experience, not by holding a ruler up to its contours. The project of phenomenology is to investigate how we make sense of the world through our subjective experiences and the conditions which make these experiences possible. A phenomenological approach calls for looking at the descriptive first-person accounts of experience that is prior to our attempt to rationalize them.

Empathy in the UX Research Process

The UX research process aims to understand the “why” behind user behavior and qualitative data is a rich source for uncovering it. In order to translate user behavior into insights that can be used for actionable design decisions, designers must be able to empathize with the personal experiences and motivations of their users. When conducting user interviews or usability tests, using a “think out loud” method for participants who are interacting with an interface allows us tap into their phenomenological first impressions. UX designers will gauge the *perception* of usability a product has and a phenomenological approach can help us be fine tuned in decoding the “whys” behind that perception. The French philosopher Maurice Merleau-Ponty investigated the conditions of human perception to uncover how our perception is colored by our bodies.

8.6 MENTAL MODELS AND CONCEPTUAL DESIGN

A mental model is a representation of the world that is created in the mind. Mental models help us to understand and interact with the world around us. Mental models can be used to represent our understanding of objects, people, or situations. Mental models can be used to solve problems, make predictions and make decisions.

Identifying and understanding mental models of people helps create solutions to their problems that are more usable and easier to learn. Often people learn through physical analogies for tasks. For example, scroll bars or scrolling with your finger on touch devices. You drag up to move the digital content up, as you would with a piece of paper sitting on a desk. Another example are recycle or trash bins on computers. Both examples match constructs in our physical space, but the actual underlying programming and processing is a reality that is completely different. Our mental model is throwing away or moving around documents with physical movements and trash bins, this helps us remember what we need to do for those tasks. The alternative would be to know how to execute code or even lower processes to do these actions.

What are Conceptual Models?

A conceptual model is an abstract, high-level model of a system. Conceptual models can be used to represent any system, from a computer system to a business process. A conceptual model is not meant to be implemented, but rather to help you understand the system and how its components interact.

From "Activity Modeling and Behavior Modeling, C.H. Kung, A. Solvberg" a conceptual model has four fundamental objectives:

1. Enhance an individual's understanding of the representative system.
2. Facilitate efficient conveyance of system details between stakeholders.
3. Provide a point of reference for system designers to extract system specifications.
4. Document the system for future reference and provide a means for collaboration.

Although conceptual models are not meant to be implemented, they play an important part in the system's development. It can clarify system properties, requirements, and events that need to be addressed.

Mental Models vs Conceptual Models

Now that we know what mental models and conceptual models are, let's take a look at the differences between them. Mental models are used to understand objects, people, or situations. They can be used to solve problems and make decisions. Conceptual models are abstract, high-level

models of systems. Conceptual models help you understand how the system's components interact with each other.

The UX design process

Mental models are more specific than conceptual models and provide more detail about the object or situation being modeled. Mental models can be implemented while conceptual models cannot.

When should you use Mental Models in UX Design?

Mental model should be used when understanding user behavior is important. When designing an interface, it is important to understand why users do what they do. Mental models can help you with this by providing a framework for thinking about user behavior and how it will impact your design. Mental model also works well in situations where the solution is not known, but there is an issue that needs to be solved.

Mental models are also useful when users are having problems with a task in your design. By discovering how your users think about the task they are trying to accomplish you can better fit your UI and workflow to match their mental model. These should be real users, not internal stakeholders unless they are your actual users.

When should you use Conceptual Models in UX Design?

Conceptual models are used when designing interactions between components of systems. For example: conceptual model would be useful if we were building a chatbot or some other system that had more than one component (i.e.: multiple modules). Conceptual modeling helps us think through all of the different flows and possible paths while keeping each part separate from each other, which makes things easier to maintain and update.

Conceptual models are a good tool when dealing with a complex system that has "a lot of moving parts" or interactions between components. The model can provide a nice short hand for quickly understanding what requirements are needed in a project and to find gaps. If you are helping design a conceptual model, be sure to involve stakeholders and other team members that are knowledge experts across the model. They will help catch gaps or misunderstandings in the model.

8.7 WIREFRAME

Wireframing is a process where designers draw overviews of interactive products to establish the structure and flow of possible design solutions. These outlines reflect user and business needs. Paper or software-rendered wireframes help teams and stakeholders ideate toward optimal, user-focused prototypes and products.

Wireframes are basic visual guides in which designers propose elements for screens and webpages and show how experimental solutions would flow for target users. Wire framing is invaluable early in the interaction design process for design teams to explore how concepts accommodate

user and business needs. You mark out a solution’s bare bones and include navigation features and more detail than in sketches. Good wireframing is the skill of creating **realistic-looking, lean layouts** so your team and stakeholders can quickly determine if concepts are worth developing. Wireframing is distinct from prototyping in the sense that prototyping deals more with testing interactivity and—when done at the highest level of fidelity—sophisticated versions that might closely resemble the released products. However, it’s similar in that you can also do wireframing by hand (e.g., using boxes and lines to represent pictures, text, etc.) or with software and make low- to high-fidelity versions. In low-fidelity wireframing, you use placeholders to mark content and pictures in grayscale. In high-fidelity wireframing, you introduce more realism, including pictures and perhaps even some interactivity. You can adapt well-crafted wireframes far more easily into prototypes for usability testing.

Software choices vary in price (some are free), options (e.g., click-through interactivity) and suitability (e.g., for mobile). When you do wireframing well, you can help safeguard yourself, your team and your brand against pursuing flawed solutions. Good wireframing can also support agile development as team members needn’t wait for sophisticated deliverables.

8.8 PROTOTYPING

A prototype is “A simulation or sample version of a final product, which UX teams use for testing before launch.” The goal of a prototype is to test and validate ideas before sharing them with stakeholders and eventually passing the final designs to engineering teams for the development process. Prototypes are essential for identifying and solving user pain points with participants during usability testing. Testing prototypes with end-users enables UX teams to visualize and optimize the user experience during the design process.

Engineering is expensive, and making changes to a final product is often not as straightforward as teams anticipate. So, finding and fixing errors during the design process is critical!

Prototypes have four main qualities:

- **Representation** — The prototype itself, i.e., paper and mobile, or HTML and desktop.
- **Precision** — The fidelity of the prototype, meaning its level of detail—low-fidelity or high-fidelity.
- **Interactivity** — The functionality open to the user, e.g., fully functional, partially functional, or view-only.
- **Evolution** — The lifecycle of the prototype. Some are built quickly, tested, thrown away, and then replaced with an improved version (known as “rapid prototyping”). Others may be created and improved upon, ultimately evolving into the final product.

Another common misconception about prototyping is that it only needs to be done once or twice at the end of the design process—not true.

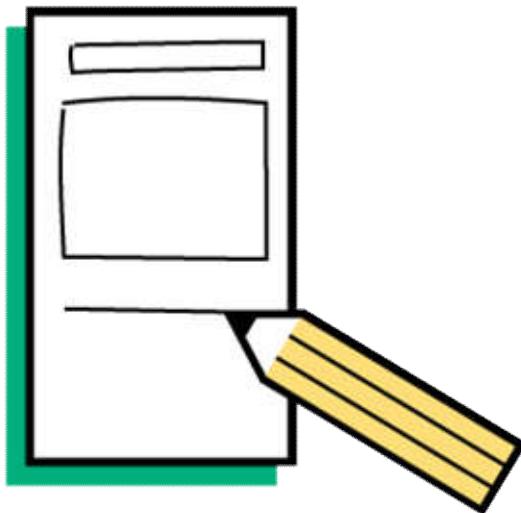
The UX design process

The Most Useful Prototyping Methodology

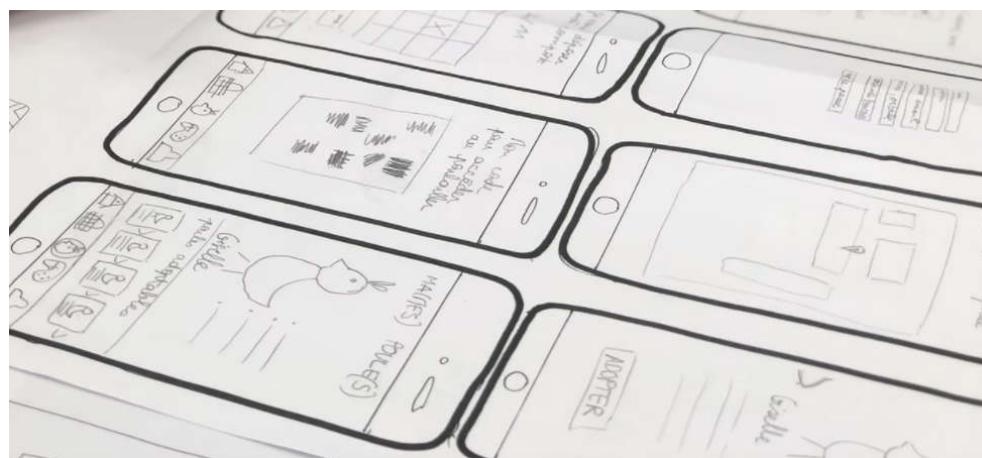
We're going to explore prototypes in three types of prototypes: paper, digital, and HTML.

Paper Prototyping

Paper prototyping works best during early design stages where UX teams collaborate to explore many concepts fast. Team members sketch ideas by hand using simple lines, shapes, and text. The emphasis is on lots of ideas and speed, not aesthetics.



UX Teams lay paper screens on the floor, table, or pinned to a board to simulate user flows. A common practice for testing these prototypes is to have one person play “the product,” switching the sketches according to how the real user behaves.



A low visual/low functional paper prototype.

Advantages of Paper Prototypes

- **Fast** — You can sketch a prototype in minutes, which is why paper works so well for testing lots of ideas. You can draw a prototype quickly (even during a brainstorming meeting), so you haven't wasted more than a few minutes if an idea falls flat.
- **Inexpensive** — You only need a marker pen and paper to create prototypes, making the process cheap and accessible.
- **Team-building** — Paper prototyping is a collaborative effort, and often teams have fun coming up with fresh ideas. It's a fantastic team-building exercise, and these free-thinking sessions often inspire creativity.
- **Documentation** — Team members can keep physical copies of paper prototypes, notes, and todos for quick reference during future iterations.

Disadvantages

- **Unrealistic** — No matter how skilled the art or craftsmanship, paper prototypes will never be more than hand-drawn representations of a digital product. So, while they're quick to draw, paper prototypes produce little or no results when doing user testing.
- **False positives** — Sometimes, paper prototypes don't validate ideas properly. What seems like a good idea on paper might not work effectively in a digital wireframe.
- **No gut reactions** — Paper prototypes rely on the user's imagination, adding a break between seeing the stimulus and responding to it. That "gut" reaction is crucial for a successful UX.

Considering these advantages and disadvantages, we recommend paper prototyping *only* during early-stage design. Once you move from paper to digital, there shouldn't be any reason to revisit hand-sketched prototypes for the same designs or user flows.

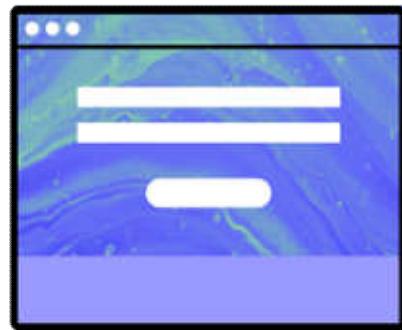
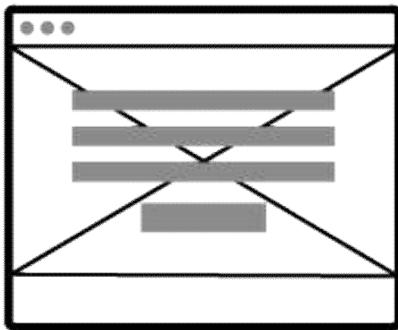
For more information on paper prototyping, check out these helpful resources:

- *Paper Prototyping as a Usability Testing Technique* by Justin Mifsud
- *Better Use of Paper in UX Design* by Marcin Treder
- *iPhone User Interface Design, Paper Prototype Design* (video)
- *Printable Paper Prototyping Templates* courtesy of Tripwire Magazine

Digital Prototyping

The UX design process

Digital prototyping is an exciting part of the design process. Prototypes start to resemble the final product allowing teams to test and validate ideas.



There are two types of digital prototypes:

- Low-fidelity prototypes: a user flow using wireframes
- High-fidelity prototypes: a user flow using mockups

Low-fidelity prototypes allow research teams to outline basic user flows and information architecture. High-fidelity prototypes go into more detail, testing user interfaces, interactions, and how usability participants interact with a product.

Designers build prototypes using design tools like Figma, Adobe XD, and others. Sometimes non-designers, from product teams use Powerpoint or Google Slides to simulate user flows.

UXPin is unique because it allows designers to create prototypes that look and function exactly like the final product—something you cannot achieve with other popular design tools!

Advantages of Digital Prototyping

- **Realistic interactions** — Testing with high-fidelity digital prototypes lets UX teams see how users interact with the final product, thus effectively iron out any usability issues.
- **Flexibility** — *Test early and test often!* You can start with lo-fi prototypes that become progressively more advanced as the product design process moves forward.
- **Speed** — While paper prototypes might be the fastest way to test ideas, digital prototypes are the quickest way to test usability issues. Once a product gets to the engineering stage, changes cost significantly more time and money.

Disadvantages

- **Learning curve** — Before you can build a prototype, you'll need to learn and understand the software—which is why product teams often use Powerpoint instead of a specialized design tool. The good news is that most design software incorporates the same tools, so it's relatively easy to switch between them.
- **Cost** — As you move from low-fidelity to high-fidelity prototyping, time and labor costs increase.



UX EVALUATION AND GOALS

Unit Structure

9.0 Objectives

9.1 Introduction

9.2 UX Evaluation and improves

9.3 UX Goals

9.4 Formative Vs Summative

9.5 Types of formatives

9.6 Informal summative evaluation methods

9.0 OBJECTIVES

Usability evaluation **helps us to gather ideas about the current design issues**. They help us to reflect on their needs and experiences. Meanwhile trying to make sense about the context, stimuli, environment factors, time factors and much more.

9.1 INTRODUCTION

A heuristic evaluation is a way to test whether a website is user friendly. In other words, it tests the site's usability. Unlike user-testing, where the site (or prototype) is evaluated by users, in a heuristic evaluation the site is evaluated by usability experts. That's why you'll sometimes find it referred to as an "expert review".

Here's a concise summary for you.

- 1. Visibility of system status:** The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
- 2. Match between system and the real world:** The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
- 3. User control and freedom:** Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

- 4. Consistency and standards:** Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
- 5. Error prevention:** Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
- 6. Recognition rather than recall:** Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
- 7. Flexibility and efficiency of use:** Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
- 8. Aesthetic and minimalist design:** Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
- 9. Help users recognize, diagnose, and recover from errors:** Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
- 10. Help and documentation:** Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Using these established guidelines for best-usability-practices, you can either **design your own set of review questions** or find some that have already been written and tweak them to your liking.

How to conduct a heuristic evaluation

In companies which have the resources, it's recommended that the site be tested by at least three usability experts. After a set of guidelines are agreed upon, each usability expert reviews the site separately. Nielsen recommends having someone who is familiar with the site **act both as a recorder and be able to answer any clarifying questions** the reviewer may have of the site, increasing the efficiency of the review.

Afterwards, you'll want to compile, review and prioritize the data from the three reviewers. The benefit of having multiple reviewers is that although they will likely catch many of the same errors, they each will likely find some the others have missed. And the number three is important because,

although with more people reviewing the site you may find more errors, UX Evaluation and Goals it's not likely to be statistically significant.

3. What if you can't afford to hire usability experts?

Freelancer designers or those working in nonprofits or small start-ups may not have the luxury of hiring 3 usability experts. Is it possible to do a heuristic evaluation on your site yourself? In her book, **The User Experience Team of One**, Leah Buley describes a similar but slightly more informal method of reviewing the usability of a site called Heuristic Markup that could come in handy.

In a heuristic markup, you set aside several hours to walk through the product yourself. Take yourself through the product, from beginning to end, as you think a user might do. You might try using one of your personas and/or thinking of the journey your users may take through the site as they try to accomplish specific tasks.

4. What's the difference between heuristic evaluation and user testing?

A heuristic evaluation can be used at any stage of a site's development, including in the early stages when developing paper prototypes. Nielsen recommends using it in conjunction with user testing. Administering the heuristic evaluation before user testing allows many of the 'obvious' errors to be caught before engaging in time-consuming and expensive user testing. Both will largely uncover different insights and errors to be corrected. Ideally, you would want to do both at several different stages of development. As the more obvious problems are discovered and solved the less-odious ones will be easier to spot and correct.

Advantages and Disadvantages of a Heuristic Evaluation

A heuristic evaluation should not replace usability testing. Although the heuristic relates to criteria that affect your site's usability, the issues identified in a heuristics evaluation are different from those found in usability testing.

Advantages:

- Uncovers many usability problems and significantly improves a product's UX
- You can obtain feedback early in the design process.
- Heuristics can help the evaluators focus on specific problems (i.e., lack of system feedback, poor discoverability, error prevention, etc.)
- Evaluating designs using a set of heuristics can help identify usability problems with specific user flows and determine the impact on the overall user experience.

Disadvantages:

- It requires knowledge and experience to apply the heuristics effectively.
- Trained usability experts are sometimes hard to find and can be expensive.
- The value of issues uncovered by evaluators is limited by their skill level.
- At times, a heuristic evaluation may set off false alarms: Issues that would not necessarily have a negative effect on the overall UX if left alone are sometimes flagged to be fixed.
- Heuristic evaluation is based on prejudged notions of what makes “good” usability.

9.2 UX EVALUATION AND IMPROVES

Design

“User Experience Design” is often used interchangeably with words such as “User Interface Design” and “Usability.” However, while usability and user interface (UI) design are essential elements of UX design; they are subsets of it. UX design often encompasses a wide variety of other fields.

A UX designer is concerned with the entire process of product acquisition and integration, including branding, architecture, usability, and functional aspects. It starts before the products are even in possession of the user.

Thus, products that have great user experience (for example, the iPhone) are designed with not just the use or consumption of the product in mind, but also the process of purchase, ownership, and even troubleshooting.

Similarly, UX designers are not only focused on developing functional products; they are also focused on other aspects of the user experience, such as pleasure, efficiency, and fun. As a consequence, no single concept of a good user experience exists. Instead, successful user experience is one that meets the needs of a specific user in a particular sense where he or she uses the product.

How to Enhance site User Experience Design

1. Everything Begins With User Research

Before you begin designing product experiences, you may be tempted to start with the question: “what exactly will it do?” However, you might just be getting ahead of yourself. In the context of UX, no question matters more than, “Who is it for?”

That’s where you start the design of your user interface. You need to know who will use your app (and anyone else who might be in your audience) long before you sit down to draw things out. What would they like? Are

their needs being met? What can your mobile app give them that they don't get somewhere else? What would you do better than your competition if they get it elsewhere?

UX Evaluation and Goals

(Hint: if you offer a better user experience, that's a serious competitive advantage.)

This doesn't mean that you have to spend thousands of dollars on market research and analysis. Find out who your target audience is, first. If you are creating a mobile experience to complement an existing one, such as a web application or a retail experience with brick and mortar, then this part is simple for you, since you already know and can survey your (current) users.

If you're creating a mobile app or webpage you can find your target audience persona by asking questions such as:

- What demographic features are you targeting? Women 19–28, men and women over 50, or maybe professionals, students, parents, etc.?
- What interests are you targeting? Gamers, volunteers, fashionistas, foodies, musicians, travelers?
- What products, apps, or services are they already using? What reviews on app stores and social media are they giving such goods and services? How do these creators respond, and what can you do better?

Your UX design process should be based on all these information. To learn as much about your target audiences and what they need as possible, use polling, surveys, statistics, and even good old fashioned Google. Remember: it's a lot easier to build experiences that people want than to make people want experiences that you have created. So, brace yourself to become a stalker:) By creating a sense of visual hierarchy, white space helps to guide users through a website and draw their attention to the most important information," says Milosz Krasinski.

2. Simplicity- Keep It Simple and Practice Responsive Design

In a website layout, simplicity essentially means getting rid of unnecessary components in a design. Note that you want a particular action to be taken by a visitor coming to your site. In order to get what they have come to search, they will scan the content. It's important to remember that if they have to navigate a maze to find what they want, they won't hang around. This is why it's imperative you get timely and contextual feedback on your UI elements as you build it. It not only helps you get better feedback but also makes sure you're on the right track.

Simplicity entails having a clean layout, a two or three-color scheme and plenty of white space, an average of two fonts, and the third one for your logo. Your graphics must have a purpose; to be both clickable and insightful.

- Every page should have a single definite goal. The checkout page, for instance, contains only what is needed for the process of checkout. Only contact information and/or a form are included in the contact list, not tips on personal-grooming habits.
- The intent of each page and each item on it is immediately understandable, without clarification, by the user. For instance, the user interface has visually distinct (more on that later) navigational elements on the page.
- Any additional but unimportant data goes to the bottom of the list. For example, with “you might also like” content, or the extra blogroll at the bottom of the post.

3. Whitespace Is Your Friend – Use It Generously

This is one of the quickest and easiest ways to enhance your design. Even subtle amounts of whitespace will allow your designs to breathe and look more polished.

The design background color does not always have to be white. It only needs to be the space between elements of the website. White space adds simplicity and elegance to your web pages.

Here are some ways you might use white space to boost the UX of your website:

- Increase the line space for your text in the body
- Increase white space in long text blocks on the left and right margins. It's been shown to improve comprehension up to 20%!
- Avoid putting images in line with text instead of positioning them on top or under text blocks.
- Group and surrounding related objects with white space

4. Make Different Elements Visually Distinct

One of the most significant goals for UX designers is a visually distinct page layout. It is a way to retain a flexible user path and engaging user experience.

Long story short: make it easy for your users to find whatever they are looking for on your pages with minimum fuss.

Additionally, make your website/app navigation visibly distinct.

Here are a few tips that might be useful during the design process:

- The most important details on the website ought to stand out the most. If it's a blog post, you need a clear-cut headline like “Best trends in

- Users must be aware of their location on the mobile app or website. There must be navigational tools at hand. For instance, at the top of the page, you have a website navigation panel with all the important parts of the website easily accessible.
- Call-to-action buttons need to stand out and provide a concise explanation of their intent. They should be readable and accessible. A subscribe icon, for instance, with a field to type in your email.
- The search field must be visually distinct with, preferably, the placeholder text “search” together with a glass icon. It’s usually at the upper right corner of a website.
- Contrast and Color: text readability and eye-friendly design are important. The readability of the text depends significantly on the colors you use and the contrast between your text color and the surroundings.

There are several tools, such as Usecontrast and Colorsafe, that will help you check the adequacy of the colors and contrast you use. Make sure color-blind users can read your website and be conscious of the contrast or general color of the mobile website.

And when it comes to colors, it’s important to be aware that:

- Generally, background colors are muted
- Blue is for text links
- Red is for important items, usually warnings or mistakes.
- Calls to action require a strongly contrasting hue that stands out from the rest.

5. Ask for Customer Feedback

Successful businesses and marketers listen to what their customers say – you should, too. If you don’t you are setting yourself up to lose out on an opportunity to better your products.

On your website, you could add a survey that asks customers to rate their experience.

Ask questions like:

- In order to enhance your experience, what should we do?
- Which features would you like to see in the future?
- Did we meet your expectations?

These questions allow individuals to explicitly tell you what they want to see on your website. You should collect the outcomes and evaluate them, searching for trends. For instance, if 70 % of respondents claim they have a problem finding specific posts on your website, this could mean that it is time to add a search feature.

Listening to input from customers and making appropriate changes allows you to improve your UX significantly.

6. Flow: Maintain Consistency of the User Flow Throughout the Journey

In terms of design, flow is when the user moves effortlessly from one aspect of the website or mobile app to another to fulfill the ultimate objective of fulfilling their goals.

Here's why it matters: a clear design framework allows consumers to get what they want seamlessly. That works to your advantage. Consistency perpetuates implementation.

You need to think through what the user is going to do step by step to keep things consistent.

Let's take the flow of a website, for example. At an entry point, usually the homepage or a blog post, the user starts their journey. You need to think about where the point of entry would lead. For instance, it may lead to another blog post that expands on the topic or a website that provides another service or product that meets the needs of the user. Dead-end pages are another thing to consider; those pages that don't lead anywhere else. You need to avoid such instances throughout the user journey when possible. Every page on your website ought to lead somewhere.

Finally, each website flow has an end target, where the needs of users and the goals of your website meet. For example, if users are searching for valuable content and you are providing valuable content, then a subscribe button or Patreon link is a reasonable end goal.

7. Page Loading Time

Your website's loading page speed is very imperative. If it takes longer than 3 seconds to load, 53 percent of people will leave your app.

Make sure that website users can quickly and easily accomplish their primary goals, without having to wait for your website to take forever to load. The load time, waiting times, and smoothness of popup animations could affect the perception of users.

8. Focus on Content

A designer's job should not only end with well-structured layouts. Do not forget about UX writing. Work in a team and demand quality material. Use the language that your users are familiar with, be user-oriented; make

The perceptions and experiences of users are significantly influenced by well-written text and effective illustrations or images. Again, instead of blindly following existing norms, you should do some little research about your users.

The user would never be drawn by an interface showing difficult and weird language or poor-quality images.

9. Make Your Website Responsive and Mobile-Friendly

More than half of all web traffic comes from mobile devices. As such, if your website isn't mobile-optimized, users are five times more likely to abandon it. So, you're basically shutting out a huge number of potential customers if you're not optimizing your website for the best mobile experience!

There's another group to remember here, aside from your clients: search engines. Google crawls both your desktop and mobile websites and uses both of them to index your website. That means that, in terms of SEO, a mobile-friendly website can help you perform better.

Here are some ways to make your website more mobile-friendly

- Make your website more responsive by providing the information you'd find on a desktop, optimized for mobile devices.
- Place buttons in the center of the screen where the thumbs of most people can access them quickly.

When designing mobile layouts, ensure users can interact with a single touch. Decide whether users can use devices with one or two hands or know the minimum size for a mobile touch target, which will help you understand the interface goals.

10. Conduct a UX Review/Audit

Naturally, as humans, we tend to focus solely on completing a task. We get our projects, we run with them, we share the results, and then we start over again. There's very little time set for reflection. That's a real shame. This encourages us to create systems filled with weaknesses and discrepancies and then continue to replicate the issues over and over again.

9.3 UX GOALS

Following are the goals of UX:

Goal 1: "I got what I need"

To give users what they need is the first goal of user experience design. Before using a product, people are mostly concerned about "**whether it is useful?" "Will this product solve my problems?"**

So a product should meet the functional demands of users first (not only those existed demands, but also potential ones). Doing user research is a good way to find out users' demands, but objectively, it's hard to measure users' needs precisely, even if it was huge company which has advanced user survey technologies.

For example, Facebook at the beginning didn't take "making friends with strangers (say, a friend's friend)" seriously as they believed that people only have curiosities on their surroundings; social network is essentially "a game among acquaintances". However the data show that most of users like to expand their circles by adding strangers as their friends. Now social platform also contributes a lot to companies and brands who want more influence online. They put money and energy on operating an official page to promote their products. All of these are unexpected demands for the early designers. Thus, collecting users' feedback constantly and make use of data or other materials to follow your users' activities is also a key to meet users' demands. If UX designers don't give users what they want, the users will give a shit.

Goal 2: "Don't make me think"

"Will I get what I want in a most simple, direct and quick way?" It would be better if you "Don't make me think".

The top-download games in app store are always those like "Don't Tap the White Tile", which people can play without brains. This shows the laziness nature of human beings. But laziness is also an important drive of technology development. As a UX designer, we have no reason to go against it unless we want to make products that are "anti-human".

How to design to give users what they want in the easiest way? First UX designers should be a mastery of the user stories & scenario of products. Which are important things that users pay much attention to? Which are secondary? How to simplify the operations by taking advantage of users' habits? As to user interface, whether the flat design method should be adopted to enable users to get most information at the first sight. Besides, the usability of products also depends on the design tools you choose. A complicated-to-use prototype/wireframe tool is a bad design itself, so how can we make good UI or product prototype with it?

Goal 3: "I really enjoy using it"

Many products have similar functions, which can all meet the users' needs to some extent. But only few of them are favored by users, why? As a music lover, I go to a concert at least once a month. Among these concerts of any scale, there are some which thrilled me from start to finish. It seems that the design of a music concert has nothing to do with product design, but as I have been immersing in design circle for many years I gradually found that an exciting concert is just like a product offering good user experience, both of them **give you the right thing at the right time**.

At an evening dinner, the starter is always delicate but of low volume. Gradually, under the influence of some spirits, the main course was served, at this moment the light is warmest and the atmosphere is the best. The ending part is usually made easy. This is a very good example to explain that “UX Designers should have a sense of rhythm (of product)”. When designing video player software, how many “ss” should the “black screen” last to draw the users’ attention, but never make them feel impatient? Why some social platforms only allow its users to access more functions after a period of time? Those are all questions that user experience designers should concern about.

Goal 4: Habit is a second nature

“Whether the product is attractive enough for me to use it for a long term?”, and even “becomes part of my life” and “makes me addicted to it”.

Facebook became a huge social platform with over 200 million users is that FB knows the mental & psychological needs of users: people win others’ attention on Facebook, which they didn’t get in real life. “Helping people to build a strong connection with external world; enabling them to follow and be followed, these are what a social platform was born for.” A functionally powerful product will no doubt be favored by people. But a product which forms a new habit has immeasurable potential. Electric light, mobile phones, new transports; Wechat, Whatsapp, these are all among the latter.

Goal 5: Make users your promoters

“Whether the product is good enough to motivate me to become one of its promoters?”

If a designer set the above 4 user experience goals when designing a product, he would be an excellent designer. The last goal, as far as I can see, is the inherent property of an excellent design: to mobilize its users. As we all know, users are the best spokesmen of your products. Companies may seek help from all kinds of resources to promote their products: KOL, famous blogger, web celebrities. However, none of them is as powerful as users. You may ask: why does product promotion has something to do with designers? If UX designers can build a relationship between the users and potential users, for example, put a “sharing on Twitter” button on the right place, there might be more people will join in (this is a method of most basic level). For another example, users need to cooperate with others when using the product (like game products). In short, to mobilize your users and make them your promoters is also an important user experience goals that good UX designers should set.

9.4 FORMATIVE VS SUMMATIVE

The purpose of **formative assessment** is to monitor student learning and provide ongoing feedback to staff and students. It is assessment for learning. If designed appropriately, it helps students identify their

strengths and weaknesses, can enable students to improve their self-regulatory skills so that they manage their education in a less haphazard fashion than is commonly found. It also provides information to the faculty about the areas students are struggling with so that sufficient support can be put in place.

Formative assessment can be tutor led, peer or self-assessment. Formative assessments have low stakes and usually carry no grade, which in some instances may discourage the students from doing the task or fully engaging with it.

The goal of **summative assessment** is to evaluate student learning at the end of an instructional unit by comparing it against some standard or benchmark. Summative assessments often have high stakes and are treated by the students as the priority over formative assessments. However, feedback from summative assessments can be used formatively by both students and faculty to guide their efforts and activities in subsequent courses.

An over-reliance on summative assessment at the conclusion of an element of study gives students a grade, but provides very little feedback that will help them develop and improve before they reach the end of the module/programme. Therefore achieving a balance between formative and summative assessments is important, although one that students don't always fully grasp and/or take seriously. Formative assessments, provide a highly effective and risk-free environment in which students can learn and experiment. They also provide a useful lead-in to summative assessments, so long as feedback is provided.

To engage students in formative assessment:

- Explain the rationale behind formative assessment clearly – make it clear to students that through engaging with formative tasks they get to gain experience with their assessments, risk-free, and can develop far stronger skills in order to obtain better grades in the summative assessments.
- Create a link between summative and formative assessment – design formative assessments in such a way that they contribute to the summative task. This lowers the workload on the students and provides them with necessary feedback to improve their final performance. An example of such assessment is producing an essay plan, a structure of a literature review, part of the essay or bibliography.
- Lower the number of summative assessments and increase the number of formative assessments – yet do not allow one single summative assessment to carry too much weight in the final grade.

The most common types of formative assessment are:

- **Quizzes** are short specific content targeted assessments that are aimed at providing data about student learning and understanding. They can quickly provide an educator with information about whether students understood the class content to help inform their future lessons.
- **Exit Slips** are typically small pieces of paper where students are asked to respond to a question in order to exit the classroom. Exit slips are purposeful to a specific lesson and specific targeted goal.
- **KWL Charts** are a graphic organizer to glean students' prior knowledge; students write what they know about a topic (K), what they want to learn about a topic (W), and what they have learned about a topic (L).
- **S.O.S** (Statement, Opinion, Support) formative assessments provide students a space to showcase their understanding of a topic by providing reasons to back up their claims; it can also be used at the start of a new topic to access prior knowledge.

These examples are considered formative assessments because they check for understanding and/or prior knowledge during a lesson. The assessment data pulled from each of these types of formative assessment allow an educator to inform their instruction and to address any student gaps either during that lesson or in the subsequent lesson.

9.6 INFORMAL SUMMATIVE EVALUATION METHODS

An *informal summative UX evaluation method* is a quantitative summative UX evaluation method that is not statistically rigorous and does not produce statistically significant results. Informal summative evaluation is used in support of formative evaluation, as an engineering technique to help assess how well you are achieving good usability and UX.

Participant

A participant, or user participant, is a user, potential, or user surrogate who helps evaluate UX designs for usability and user experience. These are the people who perform tasks and give feedback while we observe and measure. Because we wish to invite these volunteers to join our team and help us evaluate designs (i.e., we want them to participate), we use the term “participant” instead of “subject”.

Informal summative evaluation is done without experimental controls, with smaller numbers of user participants, and with only summary descriptive statistics (such as average values). At the end of each iteration for a product version, the informal summative evaluation can be used as a

User Experience Design kind of acceptance test to compare with our UX targets and help ensure that we meet our UX and business goals with the product design.

Some differences between formal and informal summative UX evaluation methods

Formal Summative UX Evaluation	Informal Summative UX Evaluation
Science	Engineering
Randomly chosen subjects/participants	Deliberately non-random participant selection to get most formative information



10

UX EVALUATION TECHNIQUES

Unit Structure

- 10.0 Objectives
 - 10.1 UX Evaluation techniques
 - 10.2 Types of evaluation data
 - 10.3 Some data collection techniques
 - 10.4 Variations in formative evaluation results
 - 10.5 Informal summative data analysis
 - 10.6 Feedback to process
 - 10.7 Evaluation report
-

10.0 OBJECTIVES

To Understand the product's value to the business. Radically improve your product's performance in a fast and cost-effective manner. Spot any UX and usability issues before spending money on new product development.

10.2 UX EVALUATION TECHNIQUES

- **Reporting Usability Test Results**

When reporting results from a usability test, you should focus primarily on your findings and recommendations that are differentiated by levels of severity. Include the pertinent information from the test plan and present just enough detail so that the method is identifiable.

- **Running a Usability Test**

Once you have planned your test and recruited your test participants, it's time to get ready to conduct your test. To do so, you'll want to

- **Recruiting Usability Test Participants**

It is vital to recruit participants who are similar to your site users for your usability testing. Depending on the site or product, you may have multiple potential users groups. Try to include representatives of all these groups or optimally, perform testing with each group separately if you r

- **Planning a Usability Test**

One of the first steps in each round of usability testing is to develop a plan for the test.

- **Usability Testing**

Usability testing refers to evaluating a product or service by testing it with representative users.

- **First Click Testing**

First Click Testing examines what a test participant would click on first on the interface in order to complete their intended task. It can be performed on a functioning website, a prototype or a wireframe.

- **System Usability Scale (SUS)**

The System Usability Scale (SUS) provides a “quick and dirty”, reliable tool for measuring the usability. It consists of a 10 item questionnaire with five response options for respondents; from Strongly agree to Strongly disagree. Originally created by John Brooke in 1986.

- **Heuristic Evaluations and Expert Reviews**

In a heuristic evaluation, usability experts review your site’s interface and compare it against accepted usability principles. The analysis results in a list of potential usability issues.

- **Eye Tracking**

Eye tracking involves measuring either where the eye is focused or the motion of the eye as an individual views a web page. Purpose of Eye tracking When your site visitors are connected to eye tracking software, you will be able to tell:

- **Contextual Interview**

During these interviews, researchers watch and listen as users work in the user’s own environment, as opposed to being in a lab. Contextual interviews tend to be more natural and sometimes more realistic as a result.

- **Focus Groups**

A focus group is a moderated discussion that typically involves 5 to 10 participants. Through a focus group, you can learn about users’ attitudes, beliefs, desires, and reactions to concepts.

- **Remote Testing**

Remote usability testing allows you to conduct user research with participants in their natural environment by employing screen-sharing software or online remote usability vendor services. In general, tests should be about 15–30 minutes long made up of about 3-5 tasks.

- **Mobile Device Testing**

Testing mobile devices such as phones, tablets, and eReaders requires special equipment and methodology. Since traditional desktop screen-

capture software cannot adequately capture touch interactions, usability practitioners have been using strategically placed cameras to record usability test i

UX Evaluation Techniques

- **Scenarios**

Scenarios describe the stories and context behind why a specific user or user group comes to your site. They note the goals and questions to be achieved and sometimes define the possibilities of how the user(s) can achieve them on the site.

10.3 TYPES OF EVALUATION DATA

Many types of evaluation exist, consequently evaluation methods need to be customised according to what is being evaluated and the purpose of the evaluation.^{1,2} It is important to understand the different types of evaluation that can be conducted over a program's life-cycle and when they should be used. The main types of evaluation are process, impact, outcome and summative evaluation.¹

Before you are able to measure the effectiveness of your project, you need to determine if the project is being run as intended and if it is reaching the intended audience.³ It is futile to try and determine how effective your program is if you are not certain of the objective, structure, programming and audience of the project. This is why process evaluation should be done prior to any other type of evaluation.³

Process evaluation

Process evaluation is used to “measure the activities of the program, program quality and who it is reaching”³ Process evaluation, as outlined by Hawe and colleagues³ will help answer questions about your program such as:

- Has the project reached the target group?
- Are all project activities reaching all parts of the target group?
- Are participants and other key stakeholders satisfied with all aspects of the project?
- Are all activities being implemented as intended? If not why?
- What if any changes have been made to intended activities?
- Are all materials, information and presentations suitable for the target audience?

Impact evaluation

Impact evaluation is used to measure the immediate effect of the program and is aligned with the programs objectives. Impact evaluation measures how well the programs objectives (and sub-objectives) have been achieved.^{1,3}

- | | |
|------------------------|--|
| User Experience Design | Impact evaluation will help answer questions such as: <ul style="list-style-type: none">• How well has the project achieved its objectives (and sub-objectives)?• How well have the desired short term changes been achieved? |
|------------------------|--|

For example, one of the objectives of the My-Peer project is to provide a safe space and learning environment for young people, without fear of judgment, misunderstanding, harassment or abuse. Impact evaluation will assess the attitudes of young people towards the learning environment and how they perceived it. It may also assess changes in participants' self esteem, confidence and social connectedness.

Impact evaluation measures the program effectiveness immediate after the completion of the program and up to six months after the completion of the program.

Outcome evaluation

Outcome evaluation is concerned with the long term effects of the program and is generally used to measure the program goal. Consequently, outcome evaluation measures how well the program goal has been achieved.^{1,3}

Outcome evaluation will help answer questions such as:

- Has the overall program goal been achieved?
- What, if any factors outside the program have contributed or hindered the desired change?
- What, if any unintended change has occurred as a result of the program?

In peer-based youth programs outcome evaluation may measure changes to: mental and physical wellbeing, education and employment and help-seeking behaviours.

Outcome evaluation measures changes at least six months after the implementation of the program (longer term). Although outcome evaluation measures the main goal of the program, it can also be used to assess program objectives over time. It should be noted that it is not always possible or appropriate to conduct outcome evaluation in peer-based programs.

Summative evaluation

At the completion of the program it may also be valuable to conduct summative evaluation. This considers the entire program cycle and assists in decisions such as:

- Do you continue the program?
- If so, do you continue it in its entirety?

- Is it possible to implement the program in other settings?
- How sustainable is the program?
- What elements could have helped or hindered the program?
- What recommendations have evolved out of the program?^{3,4}

UX Evaluation Techniques

10.4 SOME DATA COLLECTION TECHNIQUES

Data collection is an extensive procedure of gathering details about a particular topic in a systematized manner. The collected information must be such that it answers the primary questions about the topic of interest. It is done to prove hypotheses and evaluate the results.

One thing that you must always remember about data analysis is its accuracy and integrity. Irrespective of your method and approach, you need to ensure that you collect the data honestly. Only quality data can later lead to credible analysis and answer the questions at hand.

When is Data Collection Required?

You may need to collect data for academic research, ongoing projects, or an upcoming product or service. Through data collection, you can analyze the past and current information about a specific topic. You can use this analysis to formulate strategies for the future.

Organizations use data collection to analyze and learn about current trends, employees, customers, competitors, etc. The government may use it to understand and evaluate the need for formulating policies and other significant decisions.

Apart from that, it can be used in varying fields for:

- Eliminating the possibilities of errors
- Improved decision making for the future
- To implement a new plan, strategy, idea, etc

Classification of Data

Based on Data Type

Before getting to choosing the optimum collection method, you need to ensure the type of data you need for your research. It can be divided into two categories.

Qualitative Data: It refers to any data that is not numerical, or that is in the form of words, phrases, and sentences is qualitative data. These datasets usually capture the perceptions, intentions, emotions, etc., of individuals on a specific topic.

User Experience Design Some of the commonly used methods to collect qualitative data include in-depth interviews, group discussions, documents, focus groups, etc.

Quantitative Data: These include numerical datasets, which can be analyzed with mathematical techniques. Quantitative data is collected based on scales such as nominal, interval, ratio, and ordinal. Resultantly, this data is usually more accurate, reliable, and credible.

The most common data collection methods for quantitative data are surveys and questionnaires, observations, structured interviews, experiments, etc.

A yet another approach is the **mixed-method** approach that uses a combination of quantitative and qualitative data

10.5 VARIATIONS IN FORMATIVE EVALUATION RESULTS

Formative evaluation can be described for many scenarios for example in academics and work arena. A formative evaluation can be mentioned as one which happens before or during any project implementation with the motive to enhance project design and performance.

A formative assessment is far better in many ways than a summative evaluation and helps in finding out why a program doesn't work. The main hindrance for a formative evaluation is that it does not require time and money and hence it is a barrier to undertake it.

It is equally important for people to view it as a mandatory program which brings about a successful outcome.

What is Formative Evaluation?

Formative evaluation is an evaluation which is used for learning. The main intention of a formative evaluation is to present a measure to both the students and instructor and find them where they stay in their course of study.

With this measure, the instructor would be able to adjust and make a few variations so that the studies aren't affected. With the help of formative evaluations, more interest is presented by the instructor and students and their way of learning.

Why is the Formative Evaluation Important?

There are ample good reasons about the need for a formative evaluation which are penned down below. There are many chances where your project can be embarked and do not meet the real need when there is an absence of formative evaluation. Otherwise, the project can be constrained by other factors too which may not be controlled.

Formative evaluation is necessary and important for behavior change projects and community-engaged ones. This category of projects needs

formative evaluation as they are complex and requires few monitoring processes. In order to face unexpected outcomes and respond to emergent properties, formative evaluation is mandatory.

UX Evaluation Techniques

When you're unable to follow and capture successful feedback, formative evaluation can be followed as it enhances the implementation of the project. In this way, there are more chances to achieve the outcome of the project.

The various process changes are well understood with the help of formative evaluation. The actual reason for what works, what doesn't work and why such things happen are found out with the help of formative evaluation. In this way, the project can be made successful by gathering more knowledge required and enhance future project design and implementation.

Advantages of Formative Evaluation:

There are a number of advantages when the formative evaluation is considered, a few points are jotted down for your reference.

1. Develops knowledge:

The main intention of formative evaluation is that it helps in the development of knowledge and skills for the learners. With this category of evaluation, the instructors, leads or teachers are able to identify the needs of the individuals and direct them towards their objectives or educational goals.

The individual's hindrances and difficulties are found out by this method and appropriate remedies are applied to overcome them. With evaluation, the upcoming lesson or task is also planned. With formative evaluation, an assessment is offered by the instructor or teacher to make sure that the individuals have mastered the concept that has been taught to them.

2. Plans for the future:

Formative evaluation is beneficial as it plans for the future where any methods related to teaching or other career tasks can be altered. Weakness is diagnosed at an early stage and remediation is made.

By this way, the individuals are kept on track and move towards progress with continuous feedback. Future planning in case of any change in the methods of teaching or given task is planned well ahead, with formative evaluation.

3. Achieves successful outcomes:

A formative evaluation covers up a wide range of diagnostics that are required by the students or individuals. The feedback is the main parameter which enables students to reflect what they are learning and know the reason for the same. Formative evaluation assists individuals in enhancing their performance and producing successful outcomes.

4. Continuous improvement:

The other main beneficial aspect of a formative evaluation is that it is an ongoing process. By this way, the feedback is increased and issues are detected at an early stage. When academics are considered, conceptual errors are identified before they start by working with their term papers. Once they initiate with term paper they are guided and validated by their instructors with each step.

5. Provides rich picture:

A rich picture of the source or program is learned with the help of formative evaluation as it unfolds. With this type of evaluation, doors are opened for prospective learning for present program as well as future programs that are planned. The success and failure of the project can be determined with the help of formative evaluation and the reason for the same is also identified.

The complex factors that are present within the program is also identified with this evaluation. With the help of formative evaluation, the reason for the theory to be flawed or if the evaluation was deficient can be found out.

6. Provides quick feedback:

Reflective practice is assisted with the help of formative evaluation, also conflict management systems are strengthened in a number of ways and few of them are mentioned.

The major advantage is that with the help of formative evaluation, quick feedback is offered based on conflict management and resolution work capacity. For example, when a project is in process, the formative evaluation is one that offers complete feedback about the status of the project.

7. Documentation:

The next key beneficial factor about formative evaluation is that documentation is offered with this type of evaluation as to the working status of resolution and conflict management. Also, it provides with the techniques that are employed with the work issues that are identified, and impacts that are achieved in the early and middle stage of the process. This documentation is beneficial for a number of factors.

8. Setting goals and plans:

With the help of formative evaluation, planning is made and also revisions for any recommendation for plans are allowed. With this type of evaluation program implementation and program, plans are compared. There is also enhanced opportunity for reconsidering program plans and goals.

When program plans are revised and presented by means of formative evaluation, individuals who work with the program can revise plans or

also stick to new or old plans that are appropriate to present reliabilities. The formative evaluation also offers inputs for future project planning and ideas.

UX Evaluation Techniques

9. Used in complex settings:

Complex intervention in complex settings is quite difficult to be implemented. Another tough task is the managers, researchers, and others to possess a complete understanding about what must be implemented, the best strategy to be followed, elements those hinders or facilitate the process and the reason for any strategy to work or not work in an implementation process. All these queries are solved with the help of formative evaluation.

10. Complex interventions are refined:

Formative evaluations are ones that are beneficial for a number of interventions but specifically they are useful for refining wide-range and composite interventions. For example, primary care practices are ones that always implement numerous components in a concurrent manner. Hence formative evaluations are employed in such scenarios.

Disadvantages of Formative Evaluation:

With the various advantages, there are also a few disadvantages when the formative evaluation is considered.

1. Time-consuming and requires resources:

Formative evaluation is considered to be a time-consuming process if they are followed on a monthly, weekly or daily basis. These evaluations are time and resource-intensive, this is because they are in need of frequent gathering of data, analysis, reporting as well as refinement of new implementation and how effective it should be.

2. Tiring process:

Planning and exercising can be a tiring process and few recommendations cannot be implemented at all times. Hence this disadvantage leads many individuals to avoid the practice.

3. Trained and qualified professionals:

In order to process with the formative evaluation well qualified and trained individuals are required so that formative evaluation is carried over successfully and ended.

4. Develops challenges:

There are a number of methodological challenges with formative evaluation at times of rapid refinement process which takes place when trying to evaluate the impact of the intervention. Also, the measuring of outcomes is made possible only when the variant used for intervention is

implemented. Another disadvantage is the difficulty in determining the change in the intervention at certain outcomes.

5. Not appropriate for conflict resolution:

Conflict resolution needs and initiatives are not met by the formative evaluation method. Formative evaluation is a method that improves work and is also not an end by itself.

6. Funding limitations:

The intensity of formative evaluations is limited by funding aspects. If funding proposals are complete and perfect then formative evaluations can be offered with mid-course actions and corrections.

7. Results must be related to implementation:

When the formative evaluation is considered, results must be based on the context of the program implementation rather than a complete program assessment result.

A simple example can be that mid-stream evaluation can present unfavorable results, but can be made favorable by collecting summative data for prolonged periods. When considering practice, the mid-stream formative data is to be made with a lower certainty of evidence.

The evaluators should make sure that they mix up efforts with the summative approach and make usage of sample for a longer period with time so that there is a good balance between rigorous conclusion and interim feedback's.

8. Evaluators must preserve objectivity:

Intervention is shaped with ongoing feedback and where objectivity rises for evaluators. There should be a proper standard strategy format so that required distance is maintained for objectivity and at the same time detailed and formative feedback is provided.

These are a few disadvantages which must be understood by evaluators, instructors and others who handle formative evaluation. Formative evaluation is mandatory and can be focused on the disadvantages to avoid them in the future.

With all the advantages and disadvantages mentioned, the instructors or evaluators can make a note of them and make their projects or tasks successful.

9. Stages of formative evaluation:

There are three stages when the formative evaluation is considered, they are pre-implementation, implementation, and post-implementation.

The first stage in a formative evaluation is the development stage where the determinants of the current phase are identified. Along with this the

barriers and facilitators are also identified. The feasibility to access UX Evaluation Techniques proposed intervention is achieved in this initial stage.

The development stage also includes integrating findings with the intervention design as well as with refinement before implementation.

Formative Evaluation Implementation:

In the implementation stage, the implementation is the main focus where discrepancies are accessed between executions, implementation plan, and also makes an exploration about fidelity issues, exposure, and intensity. This stage is where documentation and understanding the nature of implications are achieved. The progress in the task is focused in the implementation stage. The indicators and impacts in the stage towards project goals are progressed.

The data in this stage is made use of to alternate original strategy. The implementation stage also allows positive reinforcement to a high performer and negative reinforcement to low performers.

The final stage is the post-implementation stage which is an interpretive one where the usefulness of intervention is accessed. The stakeholders' recommendations are elicited for further intervention refinements.

Satisfaction is accessed with the intervention and implementation process. In the same way, barriers are also identified in this stage. These are a few steps that are followed for formative evaluation

10.6 INFORMAL SUMMATIVE DATA ANALYSIS

When educators are instructing, it is critical for them to gauge their students' knowledge and progress toward learning goals. One way to efficiently do this is through informal assessment.

Essentially, **informal assessment** consists of quick tools to determine a student's level of understanding during the learning process. Another name for this is an informal formative assessment that measures student learning in real time, giving the teacher immediate information. This differs from **formal assessment**, which is comprised of standardized and collected data, often in the form of percentiles or standard scores, to be compared with students of the same age or grade.

Characteristics of Informal Assessments

Characteristics of informal assessment include the following:

- Performance-based
- Used to drive instruction
- Assesses knowledge of content
- Monitors a student's progress toward a goal

- Determines pace of instruction at a remedial or accelerated speed
- Grouping or scaffolding of students who need to be re-taught a skill or concept
- Supports differentiation
- Caters to an individual student's specific needs and learning styles
- Allows for immediate feedback

Informal Assessment: Examples

Informal formative assessment examples vary and are not one size fits all. The simplicity, length, or breadth of an informal assessment is dependent on the degree of information a teacher knows (or wants to know) about a student. Informal assessment examples may range from simple questions such as 'does that make sense' to more in-depth methods such as a portfolio.

While instructing, a teacher may implement an informal assessment on the spot if it is necessary to monitor a student's level of understanding in order to proceed with the lesson. However, some types of informal assessments may also be planned in advance

10.7 FEEDBACK TO PROCESS

Meaning of Feedback:

In the case of inter-personal communication, only sending of message by the sender to the receiver is not enough.

He/She must be eager to know the reaction or to get return information.

The message sent by the receiver back to the source is called feedback. It is the response by the recipient.

Importance of Feedback:

Feedback is essential in communication.

Actually, communication is a two-way process:

- (i) Sending the message, and
- (ii) Receiving the response or feedback.

The sender is interested to know the reaction of the recipient. He/She is also willing to know whether the recipient has understood the message in the same sense as intended. If there is any lapse in it, further negotiation or communication is necessary.

On the other hand, the recipient might have understood the meaning and intention of the message, but, due to barrier of some kind, say, language

barrier, he/she could not convey his/her reactions properly. In such case also actual picture is to be brought to light by removing the barriers.

UX Evaluation Techniques

In face to face communication feedback is no problem, because the speaker and the listener both are present and can ask and clarify the message instantly if there is any ambiguity. One can ask the other to repeat what has been missed. They can also observe the- gestures and facial expression in oral communication. But in the case of written communication the feedback is not instant. It requires some time.

Whatever may be the mode or channel (i.e., verbal, non-verbal or written), feedback is very important. It is the key which unlocks the success of communication. The objective of communication is bound to remain unfulfilled unless the process of communication is complete with feedback. It is an inseparable part of successful communication.

Process of Feedback:

The process or technique of feedback should include the following steps:

1. Listening and understanding the message properly.
2. Asking question if the message is not understood properly and get it clarified.
3. Understanding the message in the sense originally intended.
4. Conveying the reaction to the sender of the message.

Types of Feedback:

Feedback may be positive or negative. Positive feedback is the kind of feedback which is more or less acceptable or satisfactory to the sender. It means that the recipient of the message has responded in the way intended by the sender and taken the intended course of action. It signifies that everything is on the right track and no corrective measure regarding communication is necessary.

On the other hand, if the recipient does not understand what the sender wants to convey, the feedback must be poor and further communication becomes necessary to wipe out the misunderstanding. Thus, negative feedback suggests that the communication has not been effective and some correction, adjustment or re-assessment is required in the process.

Requirements of Effective Feedback:

Effective feedback needs to be:

1. Clear,
2. Well-timed,
3. Specific,

4. Bearing Right Attitude,
5. Truly Representative,
6. Impersonal, and
7. Informative.

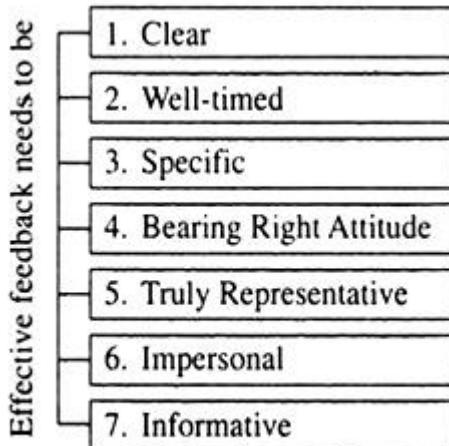


Fig. 1.7: Requirement of Effective feedback.

These points are discussed in brief in the following paragraphs:

1. Clear:

Effective feedback is possible if the recipient understands the message communicated to him/her properly.

It should be clear in two respects:

- (i) The sender of the message can understand that the feedback is related to the message sent; and
- (ii) The meaning of the feedback should be clear.

The sender of the feedback should take care about it and ensure proper understanding of the feedback by the receiver. Use of simple language for communication is highly necessary.

2. Well-timed:

Feedback may not be instant. It may require some time to get feedback. But much delayed feedback may not be worthwhile or even may not serve any purpose at all. It should be made within a reasonable time.

3. Specific:

In general, feedback, to be effective, should be particular or specific. A generalized response or an ambiguous reply is of little value. To a specific question, specific answer should be given. Sometimes general impression of a large section of people is necessary as feedback. In that case, the

communication should be designed in such a way that the people are encouraged to give their general impression. UX Evaluation Techniques

4. Bearing Right Attitude:

'I' attitude should be replaced by 'We' or 'You' attitude to get the co-operation of the person with whom communication is being made. If the sender thinks that he/she is superior to the recipient in every respect, it will be reflected in the message and the recipient will feel embarrassed in giving a feedback. Therefore, such attitude should be avoided.

The recipient should also avoid the fault-finding attitude with the sender. If much of the time is spent in finding fault with the subject-matter, presentation skill and similar matters related to the message, the feedback process is sure to suffer. Therefore, both undermining attitude of the sender and fault-finding attitude of the receiver should be avoided to make the feedback effective.

5. Truly Representative:

Feedback may be positive or negative. One should not hesitate to convey negative feedback. Actual, true and honest reaction should be communicated. We should not hesitate to convey our disagreement. We should also clearly state if we fail to understand the message.

6. Impersonal:

Feedback, to be effective, should be free from personal reactions. Let us suppose, Mr. X is depressed due to some personal problems in his life. This state of mind should not be reflected in the feedback process of the official work or business communication. Similarly, any personal rivalry or conflict should not have any influence on official feedback.

10.7 EVALUATION REPORT

Writing an evaluation report helps you share key findings and recommendations with internal and external stakeholders. A report can be used to suggest changes to how you work, to communicate your value to funders, or to share good practice with other organisations. It can also be the starting point for reporting in creative formats.

You'll need:

- data that you've collected and analysed
- an understanding of the people who'll be reading your report
- helpful colleagues to read your drafts.

Consider your audience

Think about the people you're reporting to so you can tell them what they need to know. You should consider:

- what kind of information they need (eg whether they need to know more about the difference you've made or the way in which you've delivered your work)
- how they'd like the information presented (eg as tables, case studies or infographics), and when
- why they need the information and what you want them to do as a result
- whether there are any accessibility needs that you need to consider (eg does the report need to work on a screen reader?).

Plan your report

Having a clear structure makes your report easier to read. Before you write, plan your headings and subheadings. Most evaluation reports will include the following sections.

- **Executive summary** – a synopsis of your key findings and recommendations.
- **Introduction** – a brief description of what you're evaluating, the purpose of your evaluation and the methods you've used (eg surveys, interviews).
- **Findings and discussion** – information on what you delivered, how you delivered it and what outcomes occurred.
- **Recommendations** – actions that need to be taken to respond to the evaluation findings.

Write about your findings

What to report on

Reports will vary depending on the nature of your work, but you'll probably need to include findings on the following.

- **Outcomes** – what outcomes have been achieved, for whom and under what circumstances. You should also report on intended outcomes that haven't been achieved.
- **Outputs** – what has been delivered, when and to whom. You should also report on how satisfied beneficiaries were with your outputs.
- **Processes** – information about how you delivered your outputs. You may need this information to explain why something worked particularly well, or why it didn't work.

In your report, you should **describe your data and interpret it** – analysing your data before you start writing will help with this.

Description means presenting what the data tells you. You might describe, for example, what outcomes were achieved, by whom and in what circumstances.

Interpretation moves beyond description to say what the data means – make sure you word your report clearly so the reader can tell when you're describing data and when you're interpreting it. To help you interpret data, you could do the following.

- Make connections by looking for trends, patterns and links. For example, if two groups had very different outcomes, what factors might have led to this?
- Put data in a meaningful context. Numbers don't speak for themselves. Is 70% good or bad? How do you know?

When you interpret your data, you could discuss the following.

- Why outcomes were achieved, or not achieved. Understanding this may help you make decisions about future service planning. Many funders will also want to know about this.
- What worked and what didn't. Knowing about this will put you in a good position to improve your work. It may also be useful to share with partners or funders to improve practice in the sector.
- Answers to your evaluation questions. When you planned your evaluation, you may have had two or three key questions you wanted it to answer. For example, you may have wanted to know whether your service works equally well for all groups.

Use subheadings to structure your ideas

Subheadings will make your report clear for your readers. Looking back at your evaluation framework or theory of change can help you generate ideas for subheadings. It often makes sense to have a subheading for each intended outcome.

Sometimes you'll have collected data about the same outcome from a range of different sources such as questionnaires, interviews, observation or secondary data. When you analysed your data, you probably looked at each source separately. In your report, it usually makes sense to write about all the data relating to each outcome together (rather than having separate sections on data from different sources).

Choose how to present your data

A common mistake is to try to present all your data, rather than focusing on what is most important. It helps to narrow down to what people reading your report need to know.

It's also important to think about how you'll present your information. You could consider the following points.

Which key numbers do your audience need to know?

- Decide whether to report using percentages, averages or other statistics.
- Think about whether you need to compare numerical data for different groups. You may want to look at whether men were more likely to experience outcomes than women, for instance.
- Read our guide on analysing quantitative data.

Which quotations will help you illustrate your themes?

- Choose quotations that bring your outcomes to life. Don't choose too many or they'll distract the reader from the point you want to make.
- Have a mixture of typical responses and those that don't fit easily into your categories.
- Read our guide on analysing qualitative data.

What visual aids will you use?

- Diagrams, graphs or charts should be used to highlight the most important information, rather than information which is less relevant.
- It's very easy for diagrams to mislead your audience. Here are some examples of misleading charts. If you think a diagram might be misleading, it's better to leave it out.

As far as possible, present data that has been analysed or summarised rather than raw data, to make it as easy as possible for the reader to follow.

Write accurately and clearly

It's important to write accurately and clearly so that your report can be easily understood and is not misleading.

Be transparent

Being transparent means being open about what you can and can't say, and clear about how you reached your conclusions and about the limitations of your data.

Just as it's important to minimise bias when collecting or analysing data, UX Evaluation Techniques it's equally important to **minimise bias** when reporting.

- Avoid overclaiming your role in making a difference. Your work may not be solely responsible for the outcomes that have occurred for individuals or organisations you've worked with. Remember to report on evidence of any other contributing factors (eg support received from other organisations or other sources).
- Choose case studies carefully. Evaluation case studies are not the same as marketing case studies. They should illustrate your learning points, not just the very best of what you do. You won't have a representative group of case studies, but as far as possible, choose case studies – and quotations – that reflect the full range of responses you had.
- Explore alternative interpretations or causal links. Sometimes, data is ambiguous and there could be more than one interpretation. All of us are prone to 'confirmation bias' – paying more attention to data that fits our existing beliefs. It's important to look for and talk about reasonable alternative interpretations or explanations of your data.
- Be clear about the limitations of your data. If there was a group you weren't able to hear from, or your sample over- or under-represents a particular group, say so.
- Be open about your sample size. In general, the smaller your sample, the less able you're to make generalisations about everyone in your target group.
- Report negative findings. If the data shows something isn't working or an outcome hasn't been achieved, don't ignore it. Reporting negative findings will help your audience to use the evaluation to learn and improve.

Check anonymity and consent

When you collected your data, respondents will have said whether they wanted to remain anonymous (most do) and whether you should check with them before using a quote or case study in your report. Make sure you do any checking with plenty of time before you need to complete the report.

Depending on the size of your sample and how easy it is to identify individuals, you may have to do more than just change the name to make someone anonymous. You might have to change their age or other identifying details, or remove references to anything that would allow people to identify them as an individual.

Use clear language

Evaluation reports need to be as **clear and precise** as possible in their wording. Be especially careful about using the word 'proof' or 'prove'. To 'prove' something requires 100% certainty, which you are very unlikely to

have. 'Indicates', 'demonstrates', 'shows', 'suggests' or 'is evidence for' are useful alternative phrases.

Keep your language simple and straightforward. Remember to explain any terminology that might be unfamiliar to your audience.

Develop your recommendations

Your recommendations are likely to be one of the most important parts of your report. Good recommendations will make your evaluation findings more likely to be used.

Recommendations are more likely to be implemented if the following factors are considered.

- **Supported by evidence** – be clear how the recommendations build on the key findings. It can help to structure the recommendations in the same order as the main findings to help readers understand the evidence base for each.
- **Specific** – say exactly what action needs to be taken and when within the control of the evaluation.
- **Users** – make sure individuals or groups have the authority and capability to take forward what you're suggesting.
- **Realistic and achievable** – recommendations should be feasible. You can categorise them by which ones are easy to implement and which are less so. More 'difficult' recommendations might need budget or staff changes. These should still be stated, but so should their implications.
- **Prioritised** – it's helpful to indicate some priorities for action. You could, for example, split your recommendations into 'essential' versus 'optional' or 'for consideration' versus 'for action'. Make sure the number of recommendations you include is achievable.

Involve people in the reporting process

You can involve other staff, beneficiaries and external stakeholders at several points. For example, you could share your report drafts and ask them to help you refine the conclusions. This 'co-production' of findings can be valuable and yield interpretations you may not have thought about.

You can also co-produce recommendations by sharing the findings with stakeholders and asking them to suggest and prioritise recommendations. If you do this, take care to guide people to base their recommendations on the evidence, and not their own interests or preoccupations.

Allow time for a couple of report drafts and make sure that there are people available to review the report for you. It's good to have someone look at it with 'fresh eyes'. If the report is being widely shared, you could have someone from outside your sector review the draft to make sure it's clear for external audiences.

To complete the report, leave time for proofreading and editing, checking references, and design and print if needed. You might include your data collection tools in appendices – this could help other organisations working in your field to improve their evaluation



UX METHODS FOR AGILE

Unit Structure

11.0 Objective

11.1 Introduction

11.2 Basics of agile SE method

11.0 OBJECTIVE

The goal of Agile is to produce shorter development cycles and more frequent product releases than traditional waterfall project management. This shorter time frame enables project teams to react to changes in the client's needs more effectively.

11.1 INTRODUCTION

UX design is the process by which the process flow and execution of a feature or product are conceptualized, architected, and laid out. The focus is on ease of learning and use, addressing customer needs, and accessibility. UI focuses on visual design, branding, typography, icons, and other aesthetics.

Agile UX brings Agile software development together with the product and interaction design done by UX specialists. It embeds a UX expert on the Agile team and requires understanding and valuing the UX role. This means allotting time and budget for UX's full process including research and testing.

Agile methodologies are approaches to product development that are aligned with the values and principles described in the Agile Manifesto for software development. Agile methodologies aim to deliver the right product, with incremental and frequent delivery of small chunks of functionality, through small cross-functional self-organizing teams, enabling frequent customer feedback and course correction as needed.

In doing so, Agile aims to right the challenges faced by the traditional “waterfall” approaches of delivering large products in long periods of time, during which customer requirements frequently changed, resulting in the wrong products being delivered

Application of Agile Methodology

Through most of its brief history (since 1999-2000), “Agile” has been predominantly an approach to software development and IT application development projects. Since then, however, it now extends to other fields, too, especially in the knowledge and services industries.

Agile is about being responsive to the market and to the customer by responding quickly to their needs and demands and being able to change direction as the situation demands. Be it IT or software development or any other field where there is a flow of work and delivery of work products, Agile methods are applicable. Agile methods attempt to maximize the delivery of value to the customer and minimize the risk of building products that do not – or no longer – meet market or customer needs.

UX methods for Agile

They do this by breaking up the traditionally long delivery cycle (typical of the legacy “waterfall methods”) into shorter periods, called sprints or iterations. The iteration provides the cadence for delivering a working product to the customer, getting feedback and making changes based on the feedback.

Thus, Agile methods have sought to reduce delivery times (delivering early, delivering often) to ensure that smaller vertical chunks of the product get to the market, enabling customers to provide feedback early and ensure that the product they finally get meets their needs.

Agile has become an umbrella term for a variety of planning, management and technical methods and processes for managing projects, developing software and other products and services in an iterative manner. These methods include Scrum, by far the most prevalent and popular method for software, XP (eXtreme Programming or Paired Programming), and more lately Kanban.

Agile methods also include technical practices – most of which fall under the umbrella term DevOps – that enable Test Automation, Continuous Integration/ Continuous Delivery/ Deployment (CI/ CD) and overall, an ever-shrinking delivery cycle for software and other products and services.

The use of Agile as an approach to project management has increased dramatically in recent years. Gartner predicts that agile development methods will soon be used in 80% of all software development projects.

Agile has become an umbrella term for a variety of planning, management and technical methods and processes for managing projects, developing software and other products and services in an iterative manner. These methods include Scrum, by far the most prevalent and popular method for software, XP (eXtreme Programming or Paired Programming), and more lately Kanban.

Agile methods also include technical practices – most of which fall under the umbrella term DevOps – that enable Test Automation, Continuous Integration/ Continuous Delivery/ Deployment (CI/ CD) and overall, an ever-shrinking delivery cycle for software and other products and services.

The use of Agile as an approach to project management has increased dramatically in recent years. Gartner predicts that agile development methods will soon be used in 80% of all software development projects.

Core values of Agile Manifesto

Individuals and interactions over processes and tools – The first value emphasizes teamwork and communication. We must understand that software development is a human activity and that the quality of interaction between people is vital. Tools are an important part of software development, but making great software depends much more on teamwork, regardless of the tools team may use.

Working software over comprehensive documentation – Documentation has its place and can be a great resource or reference for users and coworkers alike. The main goal of software development, however, is to develop software that offers business benefits rather than extensive documentation.

Customer collaboration over contract negotiation – Development teams must work closely and communicate with their customers frequently. By listening to and getting feedback, teams will understand what all stakeholders really want.

Responding to change over following a plan – Changes are a reality in Software development, a reality that your Software process should reflect. A project plan must be flexible enough to change, as the situation demands.

12 Principles of the Agile Manifesto

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Businesspeople and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.

- Simplicity—the art of maximizing the amount of work not done—is UX methods for Agile essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Key Agile Methodologies

Agile is an umbrella term for several methods and practices. Let's look at some of the popular methodologies:

- Scrum
- Extreme Programming (XP)
- Adaptive Software Development (ASD)
- Dynamic Software Development Method (DSDM)
- Feature Driven Development (FDD)
- Kanban
- Behavior Driven Development (BDD)

Scrum Methodology

Scrum methodology is a simple framework for working with complex projects, and it was created by Ken Schwaber and Jeff Sutherland.

Agile software development methodologies are iterative, meaning the work is divided into iterations, which are called Sprints in the case of Scrum. Scrum is executed by small teams of between 7-9 people, including a Scrum Master and a Product Owner.

In Scrum, projects are divided into cycles (typically 2 or 3 week cycles) called Sprints. The Sprint represents a timebox within which a set of features must be developed. Multiple sprints might be combined to form a Release – where formal software/ product delivery is made to the customer/ market.

The overall product functionality is broken down by the Product Owner into smaller features (typically described as Epics and User Stories – or just Stories). These Stories are prioritized and taken up in each Sprint or Iteration. The intent of the method is for the team to be able to demo at the end of each Sprint working pieces of the product to the Product Owner, to make sure that the product is working as intended.

Overall, the Scrum method breaks the long waterfall process delivery into smaller cycles, which enables product teams and the end-customer to frequently review working software and ensure that it meets their business

requirements. This ensures that the end product also meets the final requirements of the customer.

The Scrum method is characterized by specific ceremonies such as the Daily Standup meeting, the Sprint Review Meeting, the Demo to the Product Owner and the Sprint Retrospective meeting. All of these meetings provide collaboration and review opportunities to the team to ensure that development is progressing as intended, and any issues are resolved quickly.

Extreme Programming (XP)

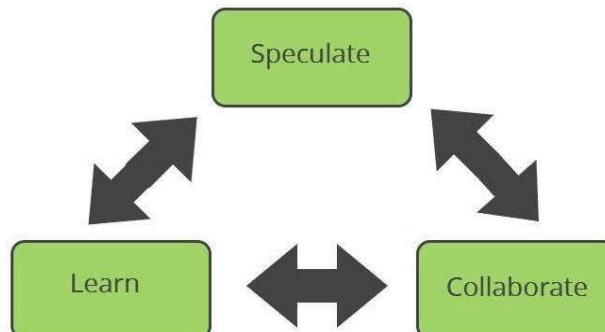
Extreme Programming (XP) – or Paired Programming is a methodology developed by Kent Beck in the early 90s. This agile methodology focuses on enhancing interpersonal relationships as a key to success in software development. XP also focuses on promoting teamwork, caring for the learning of developers, and fostering a good working environment. It is characterized by developers working in pairs where one developer programs while the other developer observes; and they switch these roles on a regular basis throughout the Sprint. This way, they enable continuous code review and feedback that enhances code quality and developer capability.

Extreme Programming (XP) promotes continuous feedback between the client and the development teams, fluid communication between all participants, simplicity in the implemented solutions and the readiness to face changes. XP is especially suitable for projects with indistinct and highly changing requirements, and where there is high technical risk.

Adaptive Software Development (ASD)

Adaptive Software Development (ASD) was developed by Jim Highsmith and Sam Bayer in the early 1990s. It incorporates the principles of continuous adaptation, i.e., *adapt to change and not fight against it*. Adaptive Software Development uses a dynamic development cycle known as Speculate, Collaborate, and Learn. This cycle is dedicated to constant learning and intense collaboration between developers and customers due to the constant change in the business environment.

Adaptive Software Development (ASD)



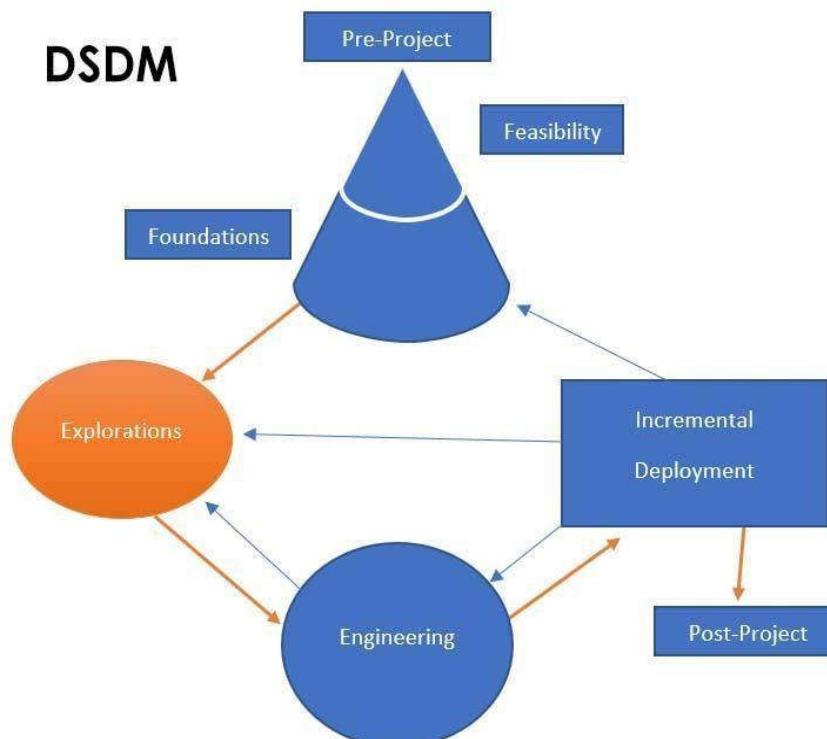
Unlike most Software development methodologies which use a static life cycle i.e., Plan-Design-Build, ASD offers a non-linear iterative life cycle, where each cycle can iterate and be modified while another cycle is being executed. It points towards Rapid Application Development (RAD), which emphasizes development speed to create a high quality, low maintenance product involving the user as much as possible. The main characteristics of ASD are:

1. Speculate: This is the initiation phase of the project where it is necessary to establish the main objectives and goals of the project by understanding the limitations (risk areas) with which the project operates.
2. Collaborate: This is the phase where most of the development is centered, maintaining co-ordination between teams that ensures what is learned by one team is communicated to the rest and does not have to be learned again by other teams from scratch.
3. Learn: The last stage ends with a series of collaboration cycles – the job is to capture what has been learned, both positive and negative. This stage is critical for the effectiveness of the project.

UX methods for Agile

Dynamic Software Development Method (DSDM)

Dynamic Software Development Method (DSDM) was developed in the year 1994 by a group of vendors and experts in the field of Software development. DSDM focuses on Software projects that are characterized by tight budgets and schedules. It focuses on frequent delivery of product cycles, and development is iterative and incremental.



With Dynamic Software Development Method (DSDM), one can design a roadmap of early and continuous deliveries for the project, implementing an incremental solution, adapting from the feedback obtained throughout the process, and checking that the expected benefits are being met.

DSDM is an agile model that can undoubtedly help organizations that are used to working on projects to change their mentality and way of working to improve their capacity to deliver value and reduce time to market.

- **Pin**
- **Share**
- **Tweet**
- **Share**
- **Share**
- **Share**

Feature Driven Development (FDD)

Feature Driven Development (FDD) methodology is mainly oriented for larger teams with more people than those to whom other agile methodologies such as Scrum are normally applied. FDD was developed by Jeff De Luca and Peter Coad in the year 1997. This methodology focuses on short iterations, which allow tangible deliveries of the product in a short period of time (2 weeks).

Projects with multiple teams and a large number of people represent the challenge that not all will be equally talented and disciplined. FDD includes specific activities that help address communication challenges and coordination of such projects.

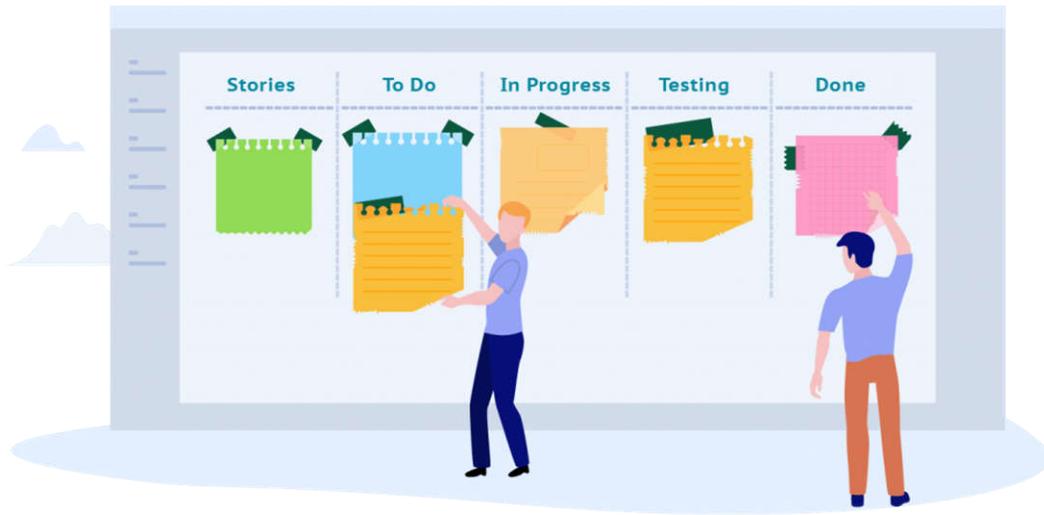
FDD is a 5-stage process, the first 3 of which are sequential and the final two stages are iterative (as shown in the diagram above). All agile methodologies follow a series of principles that make them resemble each other. FDD, however, offers solutions on how to organize the team and how to program the code, which makes it especially viable for large development teams building complex software.

Kanban Method

The Kanban Method was defined by David Anderson in the early -to-mid 2000s, in response to some of the challenges of the various Agile methods, especially Scrum. These methods, while trying to solve the challenges of traditional/ waterfall methods, became victim to some of the same challenges themselves.

The 2-3 week sprint cycle became too long to wait for many business contexts, the changes required in organizational structure (new roles and responsibilities) and a project management/ planning processes put too much strain on organizations, and many teams found themselves not

meeting even sprint-level commitments of scope and quality. For most UX methods for Agile organizations, implementing these methods became very disruptive.



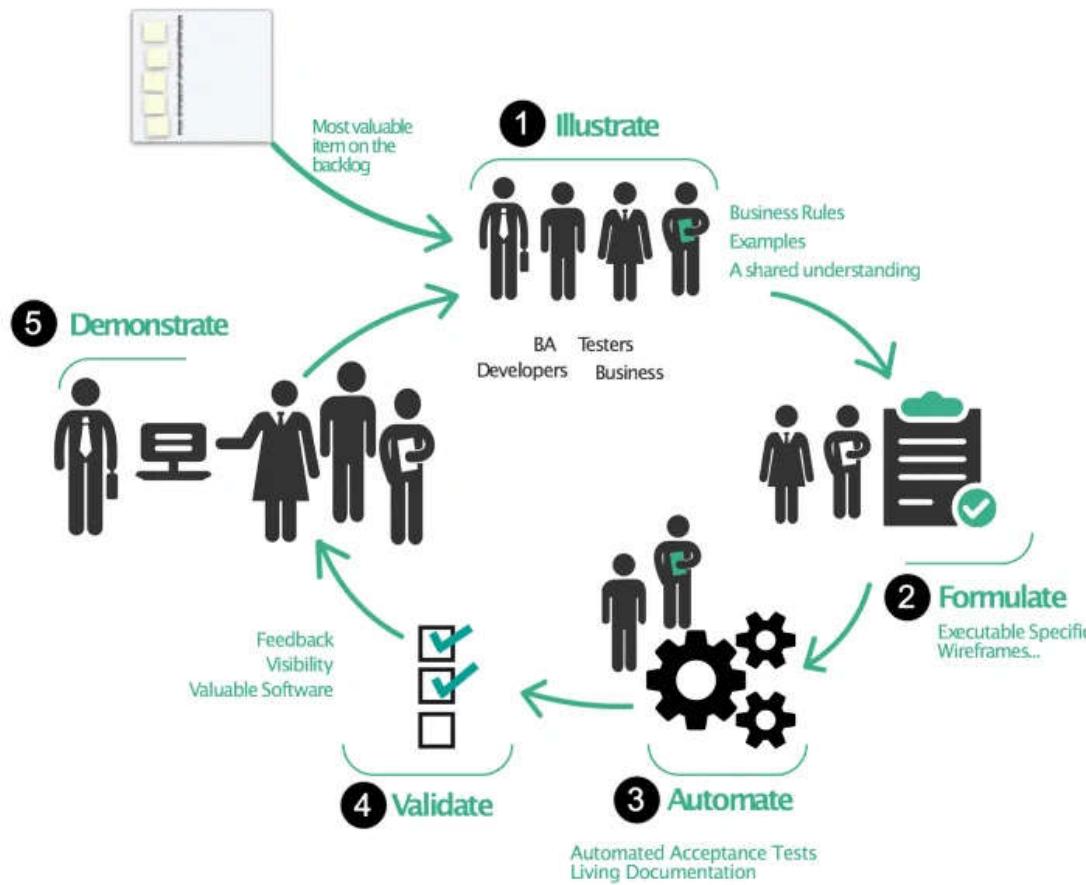
The Kanban Method was defined as the opposite of that – a non-disruptive evolutionary method for improvement, that ultimately enables teams to deliver continuously instead of in time-buckets of 2-3 weeks, get feedback faster and reduce the lead time to deliver value to the customer.

Kanban is a visual system for managing work as it moves through a process. Kanban visualizes both the process (the workflow) and the actual work passing through that process. The goal of Kanban is to identify potential bottlenecks in your process and fix them, so work can flow through it cost-effectively at an optimal speed or throughput.

Kanban is defined as a highly effective and efficient production system. The origin of the Kanban methodology lies in the “just-in-time” (JIT) production processes devised by Toyota, in which cards were used to identify material needs in the production chain.

Behavior Driven Development (BDD)

Behavior Driven Development (BDD) is a behavior-oriented agile development methodology. It was created by Dan North in 2003 as an evolution of the TDD methodology. Dan North aimed to bring non-technical people together in the process of creating the system’s technical functionality. It happens that when we develop software, we involuntarily fail to include business concepts present in the functionality, resulting in a possible flow for recurring and even serious bugs.



BDD uses universal language concepts that encourage collaboration between people with or without technical knowledge in a software project. The BDD development process is based on writing test scenarios and features. These contain the requirements and acceptance criteria for the system behavior. It tells you what the functionality needs to get started, what it will do next, and what the results will be after it is executed.

BDD helps teams more accurately communicate requirements, discover defects early, and build software that remains sustainable over time.

11.2 BASICS OF AGILE SE METHOD

Agile is a project management approach developed as a more flexible and efficient way to get products to market. The word ‘agile’ refers to the ability to move quickly and easily. Therefore, an Agile approach enables project teams to adapt faster and easier compared to other project methodologies.

Many of today’s projects have more unknowns than a traditional project management methodology can adequately handle. This uncertainty makes it challenging to document requirements and adapt to changes successfully.

The benefits of Agile project management are many, particularly for the following organizations and project types:

- Any project that evolves or does not have clear scope and requirements at the start
- Organizations that work in a fast-changing environment, such as technology
- Organizations that need to work closely with their customers and other external parties throughout the life of the project
- Companies that emphasize process and product improvement and are constantly looking to innovate
- Projects with many interdependent tasks, where the team needs to work closely and frequently communicate to ensure success
- Companies that need to create a prototype before building the final project outcome
- Projects that require rapid feedback from stakeholders about each product iteration before moving on to the next version or draft

Here are the five largest benefits of adopting an Agile method:

Continuous customer contact

Traditional project management methods generally only had the project team in touch with the customer at the start and end of the project. If customer requirements or expectations were not captured correctly in the beginning or changed over time, the project team had no idea until it was too late. With Agile, there's ongoing contact throughout the entire process and iterative deliveries to ensure your team is on track, so the end product will be exactly what the customer wants.

The ability to adapt

What if your customer told you halfway through a project that they needed a scope change? Using a traditional approach to project management, this either couldn't be accommodated or likely involved significant increases to both the project cost and schedule. With Agile, you can incorporate changes with minimal effort, no matter how far along in the project.

Faster delivery

Agile incorporates a continuous development approach that ensures your team is continuously delivering workable products. Instead of waiting for six to 12 months or longer for an end product, your client is getting a working version of the product at much shorter intervals, typically every two to four weeks.

Lower project risk

Your team is developing versions of the product regularly and getting customer feedback early on, minimizing the risk of a project failing. Breaking a large project into iterations reduces your risk of an iteration or draft failure. You're more likely to find small problems early that can be addressed quickly, rather than discovering a large issue only at the time of final testing before the end delivery. If later you encounter a problem or need to cancel the project, you'll have invested less time and money.

Ongoing innovation

Agile supports collaboration and continuous improvement, both of which can lead to innovation and the development of new products and features. Co-locating teams and having daily meetings encourages brainstorming and idea creation. Agile supports an “idea meritocracy” where the best idea wins out, no matter who it comes from. The project team, other stakeholders, and the customer can figure out functionality and features together.

When not to use the Agile project management method

Despite the many benefits of Agile, the methodology is not for every project or organization. But how do you know when not to use the Agile project management method? What are the advantages and disadvantages of Agile methodology?

Here are four times the Agile development methodology isn't the best option:

The outcome of your project is stable and well understood

Agile is intended to help reduce the cost of change and uncertainty on a project by breaking it down into iterative project management stages. However, if there's already very little uncertainty and a low possibility of change, Agile may not be the most effective approach. For instance, if you work in an industry with heavy regulations or where many of the project requirements are already known, you don't need iterative planning and multiple drafts.

Your project must produce a repeatable deliverable

By definition, a project is “a temporary endeavor with a beginning and an end, and it must be used to create a unique product, service or result.” But what if a customer asks you to construct five identical houses, and you decide to create a separate project and team for each one? Using Agile could result in five unique houses rather than five identical ones. One of the disadvantages of Agile is that it's not designed for reproducibility.

Your stakeholders don't want Agile

An Agile project requires continuous contact with your stakeholders. However, some may not have the time, ability, or desire to dedicate

themselves to a project. If the project is considered low value or low risk, they may prefer a more traditional approach where you only involve them at key phases or final delivery.

UX methods for Agile

Your company cannot support Agile

If your company or project team isn't ready, adopting the Agile development cycle can introduce risk to your project.

Here are five indicators that your company isn't ready to use Agile:

1. Agile isn't well understood. If your team and company don't have training in Agile or a sound understanding of its principles, practices, and frameworks, then they're not ready to use it.
2. Key stakeholders are resistant. Whether it's your project sponsor or a key team member, if someone is resisting adopting Agile, then you'll need to resolve the issue before you can successfully adopt the methodology.
3. Your organization cannot support daily collaboration. If there are significant barriers to communication and open collaboration between team members, Agile may not be the best approach.
4. The company structure cannot support cross-functional teams. On an Agile project, people from many different operations need to meet, communicate, and collaborate throughout the project's life. If functions are siloed in your company, this might not be realistic.
5. Your organization requires heavy documentation. If your company requires extensive documentation and test reports, Agile may be too costly to adopt. One of the 12 Agile principles is to reduce project reports, requirements, and traceability matrices.

Agile vs. Scrum

The Agile method outlines the best practices for organizing projects based on the four values and 12 Agile software development principles documented in the Agile Manifesto. You may have seen comparisons between Scrum and Agile. The definition of Scrum is “a framework for project management that emphasizes teamwork, accountability, and iterative progress toward a well-defined goal.” Scrum is a framework that you can use to implement these Agile principles, values, and best practices.

To better understand Agile vs. Scrum, you can think of Scrum as a guideline for adopting the Agile approach to project management. Scrum provides the rules, roles, events, tools, and artifacts necessary for successfully adopting an Agile mindset.

The main difference between Agile and Scrum is that Agile is the process you want to achieve, and Scrum is a tool for its success. You will often hear Scrum and Agile used interchangeably, as Scrum is the most popular

framework for Agile. However, it's not the only framework you can implement to plan and run an Agile project.

Agile vs. Waterfall project management

Agile methodology project management and Waterfall project management are two of the more popular methods for organizing projects. When determining how to plan and run your next project, you'll likely start by comparing Agile and Waterfall project management to determine which is more suitable.

Waterfall is a more traditional approach to project management, involving a linear project flow. It's best for clearly defined projects with definitive timelines and well-defined deliverables from the outset. In other words, if your major project constraints are well understood and documented, Waterfall is likely your best option.

The core principles of Waterfall include:

- Gathering all requirements upfront
- Completing all work in structured, sequential, pre-defined phases
- Conducting testing only after product development or construction is completed

Conversely, Agile was created to handle projects where the primary constraints are not well understood.

As discussed earlier, Agile is an iterative, flexible approach to projects. This method splits a project into stages or “sprints,” allowing it to mature and evolve as you gain more information.

When deciding between Waterfall project management and Agile, it should ultimately come down to your end product's maturity and how well you understand the project's outcomes and requirements in the initiation phase.



12

AGILE METHODS FROM THE UX PERSPECTIVES

Unit Structure

12.0 Objective

12.1 Introduction

12.2 Drawbacks of agile SE methods from the UX perspective

12.3 A synthesized approach to integrate UX

12.0 OBJECTIVE

Agile UX aims to bring an iterative approach to the design and improvement of features that are being built through team collaboration and the stewardship of customer feedback.

12.1 INTRODUCTION

Agile software development presumes and embraces the unknown as part of building and shipping a product. In other words, Agile views product development as an **iterative process**. On the other hand, conventional design methods stress the need for getting as much knowledge and certainty up-front regarding the product being built, and how users might use it.

However, looking deeper, you will see that combining Agile and user experience design philosophies can mean an efficient way of building great things.

The following tips may help you in introducing and integrating Agile methods to your current UX design workflow.

Learn the Basics of Agile Software Development

Since Agile UX design is somewhat of a departure from the traditional UX design methodology, it may be useful for everyone in the production team — project managers, designers, developers, and so forth — to take part in training or self-study to familiarize themselves with the fundamentals of Agile. Agile teams may throw around words such as “sprint,” “backlog,” and “burndown chart” — not rocket-science terminology by any measure — that may cause initial confusion for the uninitiated. To help you efficiently learn about Agile, and implement it quickly in your team, there are book suggestions and articles at the end of this post.

Being on the same wavelength will help your team communicate and produce great software together.

Tailor Agile Processes to Your Needs

Agile is not a one-size-fits-all methodology. In the real world, specifics and needs vary by organization. That is why there are at least 11 well-known Agile frameworks right now.

Once you have mastered the fundamentals of Agile, you should tweak and fine-tune processes to suit your needs. For example, your company's culture may not be naturally collaborative in nature (i.e. a lot of work is done in silos).

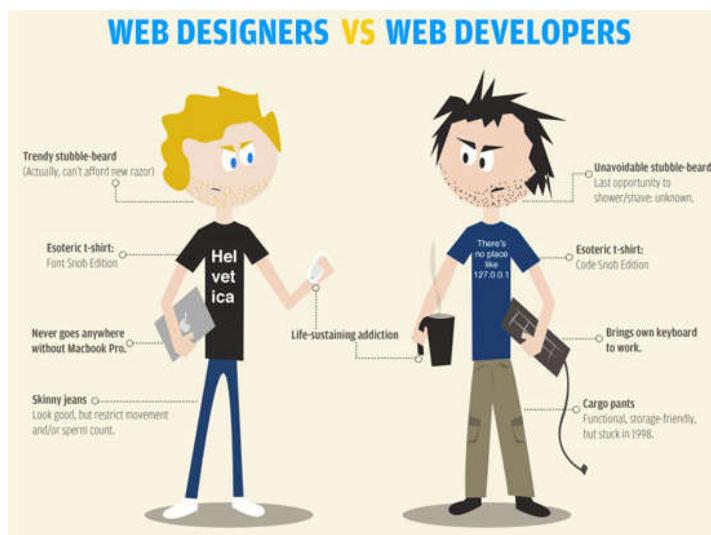
Or perhaps your team includes members who work remotely (distributed teams), which means some tweaking to the process is needed to be effective. Both these cases will present unique challenges and require adjustments.

Make a Habit of Working Iteratively

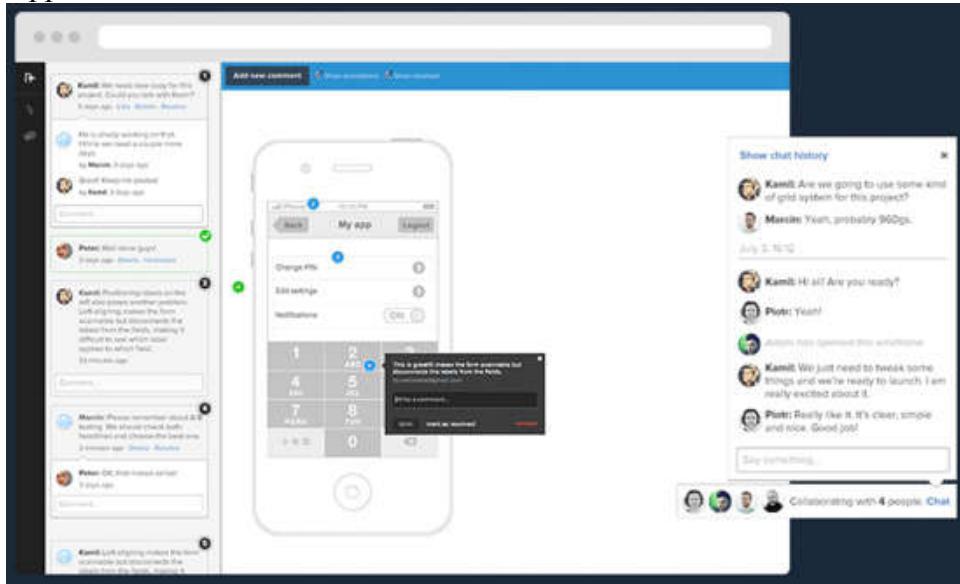
Designers by nature are taught to think bigger-picture and often strive for perfection on the get-go. Agile UX design reminds designers that product development is incremental. We should anticipate and embrace feedback and constant product changes as a catalyst for progress, as opposed to a hindrance or nuisance.

Use Collaboration Tools

Agile methods, and effective team dynamics in general, rely quite heavily on effective communication. The good news is that with the right set of tools, anything is possible... even effective collaboration between designers and developers



Explore your options when it comes to communication and project management tools such as Asana, Basecamp and Google Drive, and design platforms such as UXPin (which I am the CEO of) and Invision



Teamwork is Essential

Designers, developers, managers and stakeholders will need to collaborate throughout the design and development process as this is the key to effective Agile UX.

UX responsibilities in Scrum

Shifting from traditional product-development processes like Waterfall to modern Agile frameworks such as Scrum can be a challenge for UX. We must learn a whole new set of nomenclature, adapt to new timeframes in which to complete our research or design work, and step outside of our comfort zones to collaborate with cross-functional partners, many of whom we've never worked with before. Once we start making these changes, we quickly realize there's a lot more to Agile than simply working in time-boxed sprints.

Unlike Waterfall, Scrum has many recurring meetings that are typically referred to as ceremonies, including **daily standups** (also known as **daily Scrum**), **backlog refinement** (also known as **backlog grooming**), **sprint planning**, **demos**, and **retrospectives**. As UX people move to Agile, they may wonder whether they need to attend each ceremony and what they should do to adequately prepare and participate. In this article, I'll outline what UX should do at each Scrum ceremony to maintain open communication, influence product success, and productively contribute to the team. Much of the discussion in this article will focus on the Scrum framework for Agile, but many of the concepts can be applied to other Agile approaches as well.

Standup (Daily Scrum)

Most Scrum teams have a daily meeting that's referred to as standup or daily Scrum. (Scrum is both the name of an Agile framework and the name of a daily event on many Agile teams.) The meeting is intended to be very short, usually no more than 15 minutes, and focuses on communicating with the team to ensure that work is on track and any impediments are shared and quickly resolved. **Each member of the team, including UX, remains standing to encourage concise answers to each of the following questions:**

- **What did you do yesterday?**
 - Discuss areas of the design you worked on
 - Outline any collaboration with cross-functional partners that happened
 - Discuss any research you planned, conducted, or data analyzed
 - Share simple research findings or user insights with the team, when relevant
- **What will you do today?**
 - Keep the team aware of what you plan to do that day related to design, user research, design reviews, workshops, design sprints
 - Mention areas where you'd like input from engineering or product
 - Propose collaborative methods such as structured sketching or whiteboarding sessions
 - Invite team members to usability testing and other research happening that day
 - Hold yourself accountable to getting the work done by announcing, committing, and affirming to your group that that's what you're doing
- **What is in your way?**
 - Bring up any internal or external factors blocking you from getting work done
 - Lean on your Scrum master to help you remove these barriers to progress
 - If you need to have specific discussions with team members, suggest a quick turn-around meeting after the standup

Since UX is typically working a sprint or more ahead of the engineering team, it is already a challenge to keep the team aligned. It is for this reason that attending daily standup and clearly and concisely communicating what UX is doing are so important. Additionally, UX should listen to other

teammates during standup for any potential blockers or issues that UX may be able to address or help resolve.

UX involvement in sprint planning is important because, just like development work, UX work should be accounted for in the backlog, either with separate UX tickets that share epics with engineering, or as part of the same tickets where UX work is organized into subtasks off of the parent user story. When estimating your UX work, use t-shirt sizes, rather than story points. If your scrum team uses velocity only for engineering work, which is common. If UX work is included in the team's velocity, participate in estimating story points using the same method as your engineering team.

Most of you may have heard the terms “lean” and “agile”—or Lean user experience (UX) and Agile UX—being used to describe a team or company’s approach. You may also have been wondering what this means for you as a UX practitioner and how these approaches differ from your existing process. This guide tackles some commonly asked questions to set you up to contribute on teams using these approaches.

What is Lean UX?

Let’s start off by talking about what Lean UX means. Janice Fraser, a design and UX industry pioneer, coined the term Lean UX. She’s a founding partner of legendary design firm adaptive path and an accomplished Lean startup coach, and she defines the terms simply and clearly: “Lean UX is UX practice adapted for Lean Startups, and Agile UX is UX practice adapted for teams working with Agile.”

At the beginning of the design process, most product teams feel uncertain about which solution is the right one and need to explore various design directions before landing on the optimal solution. That’s what Lean UX is all about—it’s about formulating a hypothesis and validating it *before* committing to building anything.

Lean UX focuses on outcomes rather than deliverables. It positions UX designers as highly collaborative members of the product or service team, aiming to test their assumptions and hypotheses through the delivery of guerrilla-style user testing and experiments on minimum viable products (MVP) concepts

Adopting Lean UX means creating a culture of constant learning. Continuous discovery drives the product design process and makes product team members curious about finding better solutions.

Lean UX principles include cross-functional teams (work in teams with varied skill sets), problem-focused teams (don’t just create features, solve problems), producing the least possible waste (eliminating all work that doesn’t get your team closer to your expected outcome), and a culture of continuous discovery and improvement. See the full list of principles below.

Where did the Lean approach originate?

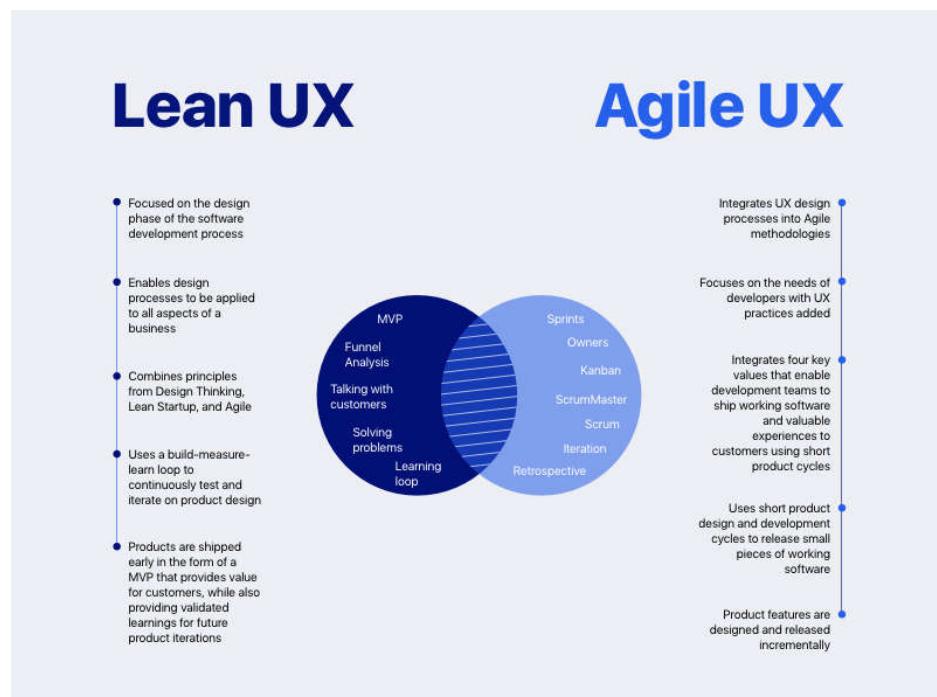
Aside from the specific UX context, it can be helpful to get a baseline understanding of Lean and Agile themselves. Lean and Agile have distinct origin stories, and the words themselves can have slightly different interpretations associated with them. While they are now practiced in an integrated way in certain software development teams, they may not necessarily go hand in hand.

Lean can refer to a few different ideas—namely, lean manufacturing, lean management, lean software development, or lean startups. In a tech and design context, we are usually referring to lean software development or lean startups.

Lean has its origins in lean manufacturing, which emerged from the car manufacturer Toyota in Japan in the late 1940s and early 1950s. In 2003, Mary and Tom Poppendieck wrote a book titled *Lean Software Development* that outlines how to apply the seven principles of lean manufacturing to developing software. In 2008, Eric Ries began outlining the Lean startup concept that applies lean management theory to startups in the technology space. His 2011 book *Lean Startup* is one of the best-selling business books ever written.

What is Agile UX?

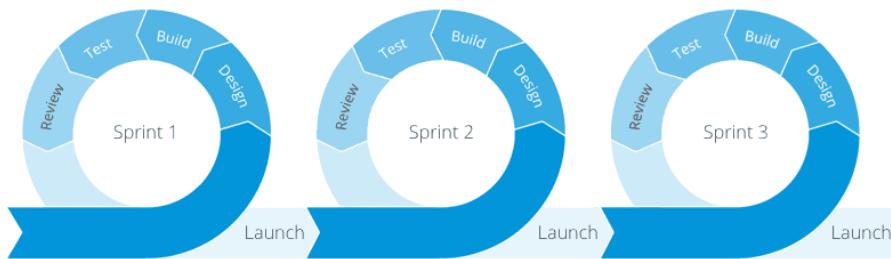
Agile UX tries to integrate UX practice with Agile software development teams. Since agile was conceived as an engineering practice, it did not originally integrate UX and design. There is much debate in the UX community about how compatible the practices are. Agile UX aims to bring an iterative approach to the design and improvement of features that are being built through team collaboration and the stewardship of customer feedback.



Where did the Agile approach originate?

Agile methods from the UX Perspectives

Agile usually refers to agile software development. A backlash occurred throughout the 1990s against heavily planned and regulated approaches to developing software with teams evolving their approaches to more lightweight and nimble ones such as scrum, crystal clear, and extreme programming. Instead of releasing everything all at once, an agile team delivers work in small increments that are easier to test.



In 2001, a group of 17 software developers came together and published the “Manifesto for Agile Software Development.” Here are a few key points from the manifesto:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

This manifesto defined Agile software development methodology as we know it today. Agile has become a catch-all term that often encompasses these approaches. It refers to a way of working that focuses on collaboration, time-boxed work blocks (sprints), and incremental evolution of a product and feature set when building software.

What do designers need to know when working with these approaches?

As companies have adopted Lean and Agile ways of operating, challenges have arisen for UX designers as they try to adapt their own human-centered design process to these approaches. Some of these challenges exist across both Lean UX and Agile UX:

- **One team, one dream.** For Agile and Lean to effectively incorporate UX, there is a need to have buy-in for the importance and value of the UX role. This requires a certain amount of organizational design maturity (both Lean and Agile have a direct connection with corporate culture, and that's why Lean and Agile thinking should be baked

in UX design system principles) as well as support at the executive level. UX design needs to be seen as an equal contributor to the process.

- **Making time for user research.** Agile software development in its origins was focused on building effectively and efficiently, so the process does not always have room for research in an explicit way. Many designers have tried to find approaches to adapt to this by doing things like starting with a Sprint 0 that allows for foundational discovery work before development starts. There is much discussion about how the design team can work in the right cadence to keep design work on pace with development.
- **Getting to the right level of documentation.** Both Lean and Agile approaches emphasize the importance of building working products rather than focusing on creating documentation. For UX designers, certain types of design artifacts are crucial to exploring and evolving the product, for example, information architecture (IA) diagrams or wireframes.

IA diagrams help product teams create meaningful content and organize it so that users can find it while wireframes are used as a reference during UI design. Some approaches for overcoming this challenge include focusing on more lightweight documentation, such as whiteboard sketches or paper prototypes.

- **Learning the language.** As with any method, Agile and Lean both have their own set of unique jargon and rituals. For Agile, understanding what is meant by scrum, sprints, the backlog, user stories, and epics will be helpful for designers. Within Lean, words like assumptions, hypothesis, minimum viable product (MVP), and pivots are some of the core concepts. Shared language and understanding between team members make up the backbone of any work.

Contributing value beats process dogma

Regardless of the methodology and approach a team decides to use, the ultimate goal for UX designers is to create value for businesses and users alike. As a UX designer on a team, the ability to collaborate, be flexible, and be adaptable will increase your chances of success. Lean UX and Agile UX give designers mindsets and models to enhance their contribution.

12.2 DRAWBACKS OF AGILE SE METHODS FROM THE UX PERSPECTIVE

Despite the benefits that Agile can offer, it's not for everyone. As a result, it's important to be aware of the disadvantages of Agile methodology. With that in mind, here are five key disadvantages of Agile.

Poor resource planning

Because Agile is based on the idea that teams won't know what their end result (or even a few cycles of delivery down the line) will look like from

day one, it's challenging to predict efforts like cost, time and resources required at the beginning of a project (and this challenge becomes more pronounced as projects get bigger and more complex).

Limited documentation

In Agile, documentation happens throughout a project, and often “just in time” for building the output, not at the beginning. As a result, it becomes less detailed and often falls to the back burner.

Fragmented output

Incremental delivery may help bring products to market faster, but it's also a big disadvantage of Agile methodology. That's because when teams work on each component in different cycles, the complete output often becomes very fragmented rather than one cohesive unit.

No finite end

The fact that Agile requires minimal planning at the beginning makes it easy to get sidetracked delivering new, unexpected functionality. Additionally, it means that projects have no finite end, as there is never a clear vision of what the “final product” looks like.

Difficult measurement

Since Agile delivers in increments, tracking progress requires you to look across cycles. And the “see-as-you-go” nature means you can't set many KPIs at the start of the project. That long game makes measuring progress difficult.

Mitigating the Disadvantages of Agile Methodology

The disadvantages of Agile methodology make clear that it's not for everyone. So what can you do?

Mitigating the disadvantages of Agile methodology requires taking more of a Lean approach by emphasizing the importance of:

- Delivering value through a quality end product rather than simply delivering a working product
- Managing a clear process for delivering that product, not an uncertain route determined along the way

Armed with knowledge about the disadvantages of Agile methodology, it's time to take a look at how Agile might affect your projects. And if you find the disadvantages of Agile outweigh the benefits, remember that there are alternative methodologies for delivering better products efficiently.

12.3 A SYNTHESIZED APPROACH TO INTEGRATE UX

Synthesis, in Human-Centred Design, is a collaborative process of sense making, which leads to creating a coherent summary of all the data

gathered during the design research. Synthesis can be explained in 3 major steps. I will present you each one by telling what it is about, why it is essential and I'll give you some practical tips on how to do it with your team.

This 1st step of design synthesis is about externalizing all the data gathered during your design research to make the content free to be moved and manipulated by the team. This way, the entire set of data can be seen at once which will help you and your team uncover implicit and hidden meanings.

WHY IT MATTERS?

Frequently designers keep the content of their research in their head or their laptop which dramatically limits their ability to see the forest for the tree. They lose the faculty to understand the research in totality and are limited in their ability to freely manipulate and associate individual pieces of data.” In Human-Centred design, the goal is to make the individual learnings group knowledge.

HOW TO DO IT?

One way of starting to make sense of your data is to share with your team the most inspiring stories you've heard from the people you're designing for. Think about user stories or experiences that have stuck with you: stories which surprised you, made you curious, or verified or falsified your assumptions. As you listen to your teammates' stories, write down notes and observations on Post-its.

By the end of this step, you will have hundreds of post-it notes on the wall organized around each research activity.

TIP 1: DECIDE HOW TO WRITE YOUR POST IT NOTES

Because of the complexity of comprehending so much data, you need to be organized. Choose a colour code that makes sense with your research, like one colour per type of research activities or users. Write one observation, insight, quote or idea per note and make sure that everyone can read them. Write concise and complete sentences which highlight an underlying behaviour and associated motivation. If it is an interview, write the name of the person.

TIP 2: START DOWNLOADING RIGHT AFTER EACH ACTIVITY

This process is best done the day of an interview or after a day in the field. Download while your experiences and perceptions are fresh.

TIP 3: SHARE ALL YOUR RESEARCH

It's essential also to share stories which falsify your previous assumptions as you don't want to risk moving forward just confirming your own wishes without really paying attention to your users' core stories and needs. Don't also forget to share surveys and secondary research.

TIP 4: IMMERSE YOURSELF

Agile methods from the UX Perspectives

For this step, you need to listen to other' stories actively, take no presumption and defer judgment

In this work the design, synthesis and optimization of a furfural production plant, considering the most abundant and common lignocellulosic wastes of Mexico is proposed. For the process, different pretreatment technologies and different purification process including intensified schemes are considered giving a total of 32 possible process alternatives. The pretreatment technologies are the dilute acid (DA) and ammonia fiber explosion (AFEX) respectively, for the separation zone we considered an azeotropic distillation process, a thermally coupled scheme distillation, a dividing wall column and one liquid- liquid extraction process. A two-stage procedure is used to determine the best process per biomass type. First, the processes are modelled in Aspen plus. Next, the best option per biomass is optimized using the differential evolution with tabu list in order to minimize the total annual cost and the environmental impact. The prescreening results indicate that the dilute acid pretreatment and the thermally coupled distillation provide the lowest cost and environmental impact for furfural production for all the raw materials.

Design is that act of problem solving—of appropriating formal qualities into a new design idea that fulfills the stated criteria and adds value to the human condition.

Software Design Synthesis Practice

Identify integrating components

At this juncture in the design synthesis practice the upper and lower tiers of the design solution have been resolved with a conceptual and fundamental arrangement of structural design elements. Between these two tiers emerges the *design chasm*, which must be bridged to unite the conceptual design layers with the structural design layers. This design chasm can only be traversed by identifying integrative structural components that either:

- Progressively extend the structural design layers toward the conceptual design components.
- Progressively decompose the conceptual design components to envelop the elements of the structural design layers.

Integrating components enable the conceptual and structural design layers to coalesce into a comprehensive, unified structural design solution. Integrating components should be recognizable as conceptual mechanisms of which the purpose is to provide an assembly packaging apparatus. The integration tier provides one or more layers of integrating components that align the lower-tier structural components with upper-tier conceptual components. The following guidelines suggest some principles for determining how to identify integrating components:

- Integrating components should provide an intuitive adaptation in the transition from the structural to conceptual design layer.
- Each integrating component should represent a significant assembly and integration action and should form a perceptible element within the structural configuration.
- Integrating components should provide a judicious progression of the software integration and testing strategy. The result of each integration and testing endeavor should be a self-contained, proven structural component with verified interfaces.
- Minor, incremental integration actions should be avoided unless they are deliberately devised to mitigate risks.

Every integrating component is a higher-level structural component. The rationale for distinguishing between the structural and integrating components is the level within the structural configuration in which they are stationed or located. Structural components are situated in the top level of the bottommost tier and were derived unambiguously from the functional architecture as large, complex functional units.

