Introduction to interface design

A user interface is the point of interaction between humans and computers. User interface *design* is the process of designing how these interfaces look and behave.

Imagine you're using an app on your smartphone to book flights. The screens you navigate, the buttons you tap and the forms you fill out are all part of the user interface.

A user interface is made up of the following elements:

- **Input controls:** These are interactive elements that enable a user to enter information. Input controls include things like checkboxes, buttons, text fields and dropdown lists.
- **Navigational elements:** These help the user to navigate an interface in order to complete their desired task. Navigational components include things like search fields, sliders and hamburger menus.
- **Informational components:** These communicate useful information to the user, for example through message boxes, notifications and progress bars.
- **Containers:** Containers are used to group content into meaningful sections. A container holds various elements, keeping them to a reasonable maximum width based on the user's screen size. An example of a container in UI design is the accordion menu—a vertically stacked list of headers that can be clicked to hide or show content.

UI design considers all of these elements and how they work together to create interfaces that are both easy to navigate and visually pleasing. As such, UI design covers:

- **Interactivity**—how the user interface and its various elements behave and function. For example, what happens when a user clicks on a particular button.
- **Visual design**—how the interface looks, considering things like colour, typography, imagery and graphics, logos, icon design and spacing. A variety of resources exist to help in this process. for example, the Flaticon website offers an extensive library of free-to-use icons and other graphics that can be implemented in your designs.

- **Information architecture**—how the content within the user interface is organised and labelled.

Understanding and conceptualizing Interface

https://olamidewilliams.medium.com/conceptualizing-a-ux-design-ee1054b0f3fa

https://uxplanet.org/ui-ux-mastery-2021-pt-2-conceptualizing-ideation-ceb835f7a5b7

understanding user's conceptual cognition

https://uxplanet.org/psychology-in-ux-design-understanding-cognitive-psychology-in-ux-design-375c940d34ab#:~:text=In%20conclusion%2C%20understanding%20the%20basics,users%20and%20enhance%20their%20experience.

Core Elements of User Experience

The 5 elements of UX design act as a guide for every designer but what are they? And how do we implement them into our work? In our guide, we explain how and why UX designers should include them in their design process.
The goal of any user experience project is to ensure users have a positive experience with a product. Yet, between the initial idea and the polished final product rests a complex web of decisions that impact the user experience in big and small ways. To provide structure and direction to the process of making these decisions – and to ensure each decision is made at the appropriate point in the process – the 5 elements of UX design act as a guide for every designer.

These 5 elements of UX design were first introduced to us by innovative user experience designer Jesse James Garrett, who wrote a book called The Elements of User of Experience. In it, he explains the steps user experience projects should go through, as well as the issues UX designers should consider at each step.

Because his book was published in 2002, well before the first smartphone was released, Garrett's ideas were specific to user experiences designed

for websites. However, his elements turned out to align so well with the process of creating smart, well-considered user-centric designs, they can be used as a framework for any project that requires a satisfying, cohesive user experience.

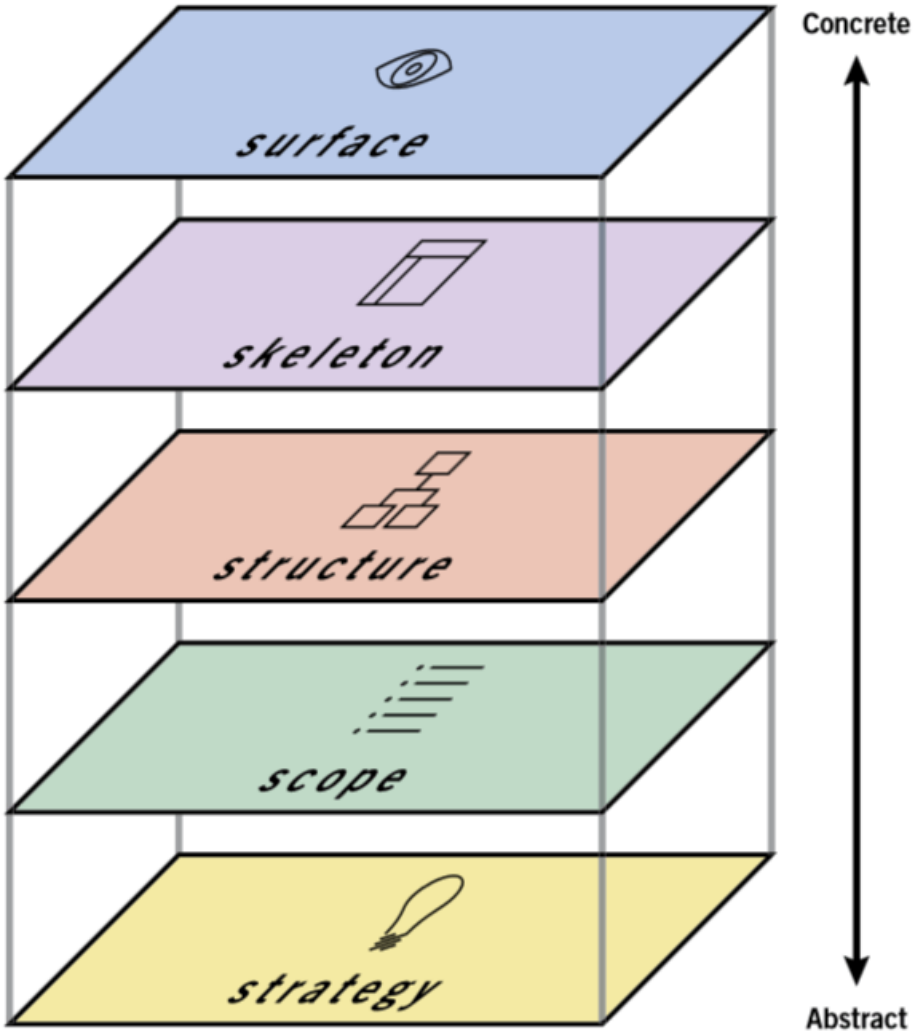
What is UX? Why has it become so important? Could it be a career for you? Learn the answers, and more, with a free 7-day video course.

Start my free course

The 5 elements of UX design

As can be seen in the diagram below, the 5 elements of user experience exist on 5 separate planes, each one stacked on top of the previous one to create an outline of the process UX teams can follow for every new project they design. From bottom to top those planes are strategy, scope, structure, skeleton and surface. Each one builds on the next as the project goes from abstract to increasingly more concrete until arriving at the finished product. In addition, each plane requires meeting different goals or completing different tasks.

Let’s look at each plane in more detail.



Strategy

The bottom plane of the model is Strategy. As the most abstract and least constrained part of the project, this is where decisions should be made about what objectives the product should be designed to meet. These objectives should include the goals that both the clients and stakeholders behind the product want to meet and the goals of the users, who will eventually look to the product to solve specific problems for them.

For example, let's say we've been hired to build an app that helps people find charging stations for electric cars. On the one hand, we must be sure to meet the product objectives, which in this case would be goals such as, "Informing electric car owners of the nearest place to charge their cars".

On the other hand, we need to meet user needs if we want users to come to our app for information. That means we need to understand what goals users would have when using the app, which we can discover through user research. In this case, we might want to learn if users would want our product to provide directions to the nearest charging station, information about how many chargers are available there and how much it costs to charge a car at each listed charging station.

Scope

After deciding on the strategy, the scope of the product can be determined and laid out in detail. It's here that all a product's features are decided upon, including the information that users can find and the functionality that users can interact with. On this plane, the UX team will create a set of functional specifications that identifies and describes every single feature of the product and a list of content requirements that identifies every single piece of content that will be included.

For instance, in our car charger finding app, on the functional specifications side we might want to include a feature to save previously discovered charging stations in our functional specifications. Meanwhile, in our content requirements we might list information like images of each charging station, maps of their locations, and details about the voltage of each available charger.

Structure

Once the scope of the product has been outlined, it's time to start working on the structure. This is where each element of navigation will be decided, including where in the product each page can be found and where users can go after arriving at a given page. This involves defining the interaction design and information architecture of the product.

On the interaction design side, we need to decide how users will interact with the site and how the system will respond, including what will happen if errors are made. This can be conveyed through conceptual models that explain each part of the user interface – usually in a flow chart format – that defines what users can do and how the product will react to each potential choice the user makes.

On the information architecture side, we need to structure the content the product offers in a way that makes it easy for users to find what they're looking for. This can be conveyed through documents like site maps that outline the hierarchy and pattern of each part of the product.

For example, to convey the structure of our electric car charger finder app, we might create a site map that shows the hierarchy of the product. This could include a home page where users can enter a location to find car charging stations. This could then lead to a list of stations each with a link that takes users to pages for individual stations.

In addition, we could also create a user flow where we show how the system responds after a user enters their location information. It can account for what happens if the system finds nearby charging stations and if there's an error that prevents the system from successfully understanding the location information provided.

Skeleton

After deciding how the product will be structured, its skeleton can be designed. This entails deciding where the navigation and functional elements from the previous plane will go on each product page. It's here

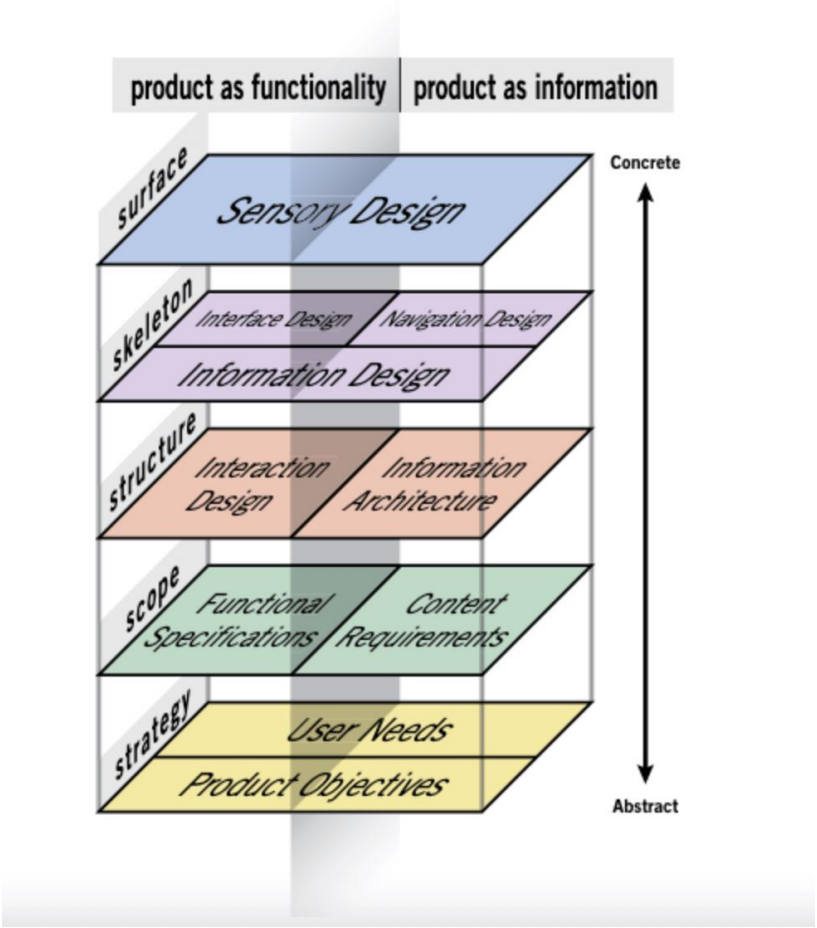
that UX designers will make decisions about the product's information design, creating wireframes and prototypes that arrange each part of the product, including the buttons, links, images and text. These are laid out in a way that ensures that users can quickly move through each page to find the information they need, while also understanding which elements of each page are interactive and which are not.

For instance, in our app, if we want to explain what users will see when they navigate to a page that describes a specific electric car charging station, we would create a wireframe that provides a blueprint of where each component of the page would go. Perhaps we have a header with the app logo and navigation back to the complete list of stations. This will be followed by an image of the charging station, followed by a link to a map of the location followed by text providing practical information about the station. This will help visualise each piece of functionality and content that will appear on the page and its placement.

Surface

The wireframes and prototypes created on the skeleton plane will be used on the surface plane – the top and most concrete plane – to create the final pages for the product. At this stage, we're concerned with the users' sensory experience. This includes how the colours and textures employed in the visual design help them understand how to navigate through and interact with the site, and how the presentation of content draws their eye to key information.

For our electric car charging app, this could mean creating a consistent colour palette and layout, where the logo appears at the top of the page, the most important information appears in a wide column in the middle of the page, and less important information is relegated to each side of the page. This layout anchors users with its consistent visual style while enabling them to easily find the information they're seeking.



While the diagram above is a neat and tidy way of laying out the 5 elements of UX design, the reality is much messier. In an ideal world the work required for each plane would be wrapped up before tackling the next one. But in real life, you'll work on more than one plane simultaneously and each decision made on one plane will impact every plane above it. As a result, you'll want to finish work on each plane before work on the plane above it is completed. This will ensure that the decisions on each plane align with one another.

What does universal design have to do with the 5 elements?

According to the Centre for Excellence in Universal Design: "Universal Design is the design and composition of an environment so that it can be

accessed, understood and used to the greatest extent possible by all people, regardless of their age, size or disability.”

This doesn’t just mean we should design products, services and environments that take account of the needs of people with disabilities or other special requirements but that we should create designs with the needs of all users in mind so that our designs are accessible to as many people as possible.

The process of utilising the 5 elements is intended to help UX designers systematically consider and account for every user need and functionality requirement of each product. This should include considering the needs and requirements of people of all kinds.

For example, the users of our car charging finder app are owners of electric vehicles, however those owners are likely to encompass a wide range of ages and abilities. As a result, in using the 5 elements as a framework for designing the app, we should consider the wide range of users who might access it. In the process, we will also be practicing universal design.

What is UX ? Give examples of Ubiquitous interaction?

1. **User Experience (UX)**: This encompasses the overall effect experienced by a user when interacting with a system, device, or product. It considers usability, usefulness, and emotional impact during interaction, as well as the memory and impressions after the interaction.
2. **Interaction with a System or Product**: Interaction encompasses various aspects, including visual perception (seeing), physical touch (touching), and cognitive processes (thinking about the system or product). It even includes the admiration of the system's presentation before physical interaction.

3. **Usability**: Usability is one of the practical aspects of UX. It focuses on the effectiveness, efficiency, productivity, ease-of-use, learnability, retainability, and the pragmatic aspects of user satisfaction.
4. **Usefulness**: Usefulness relates to the functionality of the system or product and its ability to help users achieve their goals or tasks, whether for work or leisure.
5. **Functionality**: Functionality refers to the system's capabilities and features that enable users to perform tasks or activities, whether for work or play.
6. **Emotional Impact**: Emotional impact is the affective aspect of UX, influencing user feelings and emotions. It includes elements like pleasure, fun, aesthetics, desirability, and engagement. Deeper emotional factors, such as self-expression and pride of ownership, can also be part of the emotional impact.
7. **Ubiquitous Computing**: Ubiquitous computing has expanded beyond traditional desktops and laptops, including various forms of computing devices and interfaces that are integrated into everyday life. Examples include specialized applications of robots in healthcare rehabilitation and assistance for the elderly.
8. **Specialized Robotic Applications**: Robots have evolved beyond their initial applications in housecleaning and babysitting. They are now employed in specialized areas such as healthcare rehabilitation, where they encourage interaction for severely disabled children and assist the elderly with daily tasks.

Overall, this description emphasizes the holistic nature of user experience, considering not only practical usability but also emotional and contextual aspects. Additionally, it highlights the growing ubiquity of computing in various forms, including specialized robotic applications in healthcare and other fields.

User Experience (UX): It's how you feel when you use something like a website, app, or gadget. It includes if it's easy to use, if it helps you, and if you like using it.

Interaction with a System or Product: This means everything you do with something, like looking at it, touching it, and even just thinking about it before you use it.

Usability: This is about how easy and practical something is to use. Is it efficient? Does it make sense? Can you learn to use it quickly?

Usefulness: It's about whether the thing helps you do what you want or need to do. Does it work for your tasks or goals?

Functionality: This is what a thing can do, like a tool's abilities and features.

Emotional Impact: It's how a thing makes you feel. Do you enjoy using it? Does it look nice? Does it engage you?

Ubiquitous Computing: This means that computers and gadgets are everywhere, not just on your desktop or laptop. They're in lots of everyday things like phones, appliances, and even robots.

Specialized Robotic Applications: Robots aren't just for cleaning or babysitting anymore. They're used in special areas like helping sick kids interact with their world and assisting older people with daily tasks.

Emerging desire for usability

In the past, computers were mysterious and only a few people understood how to use them. Some people thought it was okay if computers were hard to use because it made them seem important.

Nowadays, we want technology to be easy to use and enjoyable. We use it to learn, have fun, connect with others, and do good things in the world. We

don't just want it to work; we want it to look good, make us feel good, and be interesting.

People created the field of user experience (UX) to make sure technology is good quality and easy to use. But sometimes, the software industry still makes products that aren't great in terms of usability and quality. We need to do better.

The passage you've provided highlights a significant shift in the perception of computer usage and the importance of usability in technology. Here's a breakdown of the key points:

1. **Specialized Knowledge in the Past**: In the past, computers were understood by only a small group of people with specialized knowledge or a strong interest in technology. This created a sense of mystique and job security for those who understood these complex machines.
2. **Poor Usability for Mystique**: In some cases, poor usability was seen as a way to maintain this mystique around computers. If the technology was difficult to use, it reinforced the idea that only experts could master it.
3. **Aesthetic vs. Usability**: The passage contrasts poor designs that might look beautiful but are impractical to use with the desire for technology to be both aesthetically pleasing and usable. In other words, it's not enough for something to look good; it should also be functional and user-friendly.
4. **Technology's Role**: Technology is now seen as a means to learn, be entertained, connect with others, and make a positive impact on the world. Users expect more than just basic functionality; they seek

emotional satisfaction, meaning, and intellectual gratification from their interactions with technology.

5. **User Experience Concerns**: The field of user experience (UX) has emerged as a response to concerns about software product quality. It aims to ensure that technology not only works but also provides a satisfying and meaningful experience for users.
6. **Quality Concerns in Software**: The passage suggests that the software industry has room for improvement in terms of product quality. It implies that there are still concerns about the usability and overall quality of software products.

In summary, the passage highlights the evolving expectations of technology users. While computer usage was once the domain of a select few, modern technology is expected to be both functional and user-friendly. The emergence of the user experience field reflects the growing importance of providing technology that not only works but also delights and engages users.

From usability to user experience

The text you provided discusses various aspects of usability and user experience (UX). Here's a breakdown of the key points in simpler terms:

1. **Traditional Concept of Usability**: Usability is all about making sure that when people use computers or technology, it's effective (it works), efficient (it doesn't waste time), and satisfying (people like using it). It includes things like how easy it is to use, how well it helps you do your tasks, and whether you're happy with it.
2. **Misconceptions about Usability**:
    - Usability isn't just about being "user-friendly." It's about creating a tool that's efficient, effective, safe, and maybe even good-looking and fun.

    - Usability is not the same as emotional impact. While it includes user satisfaction, it also covers objective measures like efficiency and error rates.
3. **Functionality vs. User Experience**: Sometimes, a product that gives a great user experience can be more successful than one with more features. For example, the iPhone outsold the Blackberry, even though it had fewer functions.
4. **User Experience Is Not Just High-Tech**: A good user experience doesn't always mean it's high-tech or cool. It's more about how well it helps users achieve their goals.
5. **Design Beyond Technology**: Design isn't just about technology; it's about creating things (like computer systems or everyday objects) that people can use easily and effectively.
6. **Components of User Experience**: User experience includes:
    - How easy it is to use (usability).
    - How well it helps users (usefulness).
    - How it makes users feel (emotional impact).
7. **User Experience Is Internal**: User experience is about how users feel inside when they interact with something like a system or product. It includes everything from using it to just thinking about it.
8. **User Experience Can't Be Designed**: User experience isn't something you design directly. It's something users feel while using a product. It depends on the user, the context, and how they use it. The same design can lead to different user experiences in different situations.

In summary, the text explains that usability is about making technology practical and pleasant to use, while user experience is the overall feeling and satisfaction users have when they interact with it. User experience is influenced by usability, emotional impact, and how well the product helps users achieve their goals. It's not something designers create directly but something users feel based on their interactions and context.

Explain in brief the emotional impact of user experience?

1. **Emotional Aspects**: The emotional aspects of user experience include feelings like pleasure, fun, aesthetics, novelty, originality, and overall sensations. These are the affective parts of the interaction.
2. **Varied Emotional Responses**: Users can have a wide range of emotional responses to a system or product. It could be highly emotional, deeply personal, and intimate in some cases, while in others, it might be mild satisfaction or just a slight sense of contentment.
3. **Memorable Experiences**: The most cherished moments in user experience are when it reaches a high level of emotional impact. These are the instances where a product stands out and creates a deep connection with discerning users.
4. **Beauty in Design**: Sometimes, users form a strong emotional bond with a product because of its beautifully designed user experience. It goes beyond mere functionality and usability and adds something special that resonates with users on an emotional level.
5. **Transcending Usability**: This emotional impact transcends factors like form, function, usability, and usefulness. It elevates the usage experience to a state of pure joy and pleasure, akin to appreciating well-crafted music or art.
6. **Diverse Emotions**: Real emotional impact in usage isn't limited to positive emotions like pleasure. It can also encompass a wide range of feelings, including love, hate, fear, nostalgia, and the act of recalling shared memories.

In essence, the passage highlights that the user experience isn't just about functionality and efficiency; it's also about creating emotional connections and memorable moments that users cherish. Whether through beautiful design, unique experiences, or the ability to evoke various emotions, a well-crafted user experience can make technology more meaningful and enjoyable for users.

USER EXPERIENCE NEEDS A BUSINESS CASE

The passage highlights a common issue in organizations where usability problems are addressed by suggesting training as a solution. It provides a specific example of how a large governmental organization attempts to tackle user experience issues with instructional bulletins and outlines the challenges associated with this approach.

Here's a breakdown in simpler terms:

1. **Misguided Approach to Usability**: Some people believe that if a system or product is not easy to use initially, it can be fixed through training and practice. However, this approach may not always be effective.
2. **Real-World Example**: The passage mentions a real-world example involving a large government organization that serves the public. When users (agents) interact with clients over the phone, they sometimes have to follow complex procedures, which can lead to errors and missed legal requirements.
3. **Complex Procedures and Errors**: The procedures are complicated, and there are many opportunities for errors. If agents don't follow the instructions precisely, clients might not receive the required legal notices. This leads to inefficiency, risks, and increased costs.
4. **Training as a Substitute**: Using training to compensate for usability issues is costly and often doesn't achieve the desired results, such as increased productivity and reduced errors.
5. **Questioning the Approach**: The passage questions the logic of relying on such methods and instructional memos. It highlights how individuals within the organization might have become accustomed to these practices over time and see them as the norm, even if they are not the most effective solutions.

In summary, the passage underscores the limitations of using training to address usability problems and points out that this approach can be costly and ineffective. It also raises questions about why organizations persist with such methods when there are more efficient and user-friendly alternatives available.

ROOTS OF USABILITY

The roots of usability can be traced back to various fields and principles that have contributed to its development over time. Here are some key roots of usability:

1. **Human Factors Engineering (HFE)**: Human factors engineering, also known as ergonomics, focuses on designing systems, products, and environments that are safe, efficient, and comfortable for human use. It plays a foundational role in usability by considering human capabilities and limitations when designing interfaces and interactions.
2. **Cognitive Psychology**: Cognitive psychology studies mental processes like perception, memory, and problem-solving. It provides insights into how users process information, make decisions, and interact with technology. Usability design often draws from cognitive psychology principles to create user-friendly interfaces.
3. **User-Centered Design (UCD)**: User-centered design places the user at the center of the design process. It emphasizes involving users throughout the design and development stages, conducting usability testing, and iterating on designs based on user feedback.
4. **Human-Computer Interaction (HCI)**: HCI is an interdisciplinary field that explores how humans interact with computers and technology. It encompasses research, design, and evaluation of user interfaces. Usability is a core concern within HCI.
5. **Information Architecture**: Information architecture focuses on organizing and structuring information to make it more accessible

and understandable to users. It plays a critical role in the usability of websites and information systems.

6. **Usability Testing**: Usability testing involves observing users as they interact with a product or system to identify usability issues. This practice has roots in experimental psychology and user feedback, helping designers refine their designs for better user experiences.
7. **Accessibility and Universal Design**: The concept of universal design aims to create products and environments that are usable by people of all abilities. Accessibility standards and guidelines, such as the Americans with Disabilities Act (ADA), contribute to the usability of products for people with disabilities.
8. **Industrial Design**: Industrial design principles, including form and function, influence the physical design of products and how they are used. These principles have been extended to digital interfaces, impacting the aesthetics and usability of software and hardware.
9. **Usability Standards**: Over time, organizations and industries have developed usability standards and guidelines. For example, ISO standards provide a framework for evaluating and improving the usability of products.
10. **Iterative Design**: The iterative design process, which involves making multiple design iterations based on user feedback and testing, is fundamental to usability. It allows for continuous improvement and refinement of usability aspects.

In summary, usability draws from a rich history of principles and disciplines, including human factors engineering, cognitive psychology, user-centered design, HCI, information architecture, and accessibility. These roots have shaped the field of usability into what it is today, emphasizing the importance of creating products and systems that are efficient, effective, and user-friendly.

It is a matter of debate exactly when computer usability was born.
Human computer interaction in general and usability in particular owe much of their origin and development to influences from many other related fields.

Human Factors and Industrial and Systems Engineering

Human–computer interaction is clearly about human behavior and is used to drive system design, and human performance is the measurable outcome in using those systems.
We agree with all but the conclusion that the human is the most likely cause of errors or system failure; the whole point of human factors engineering is to design the system to take into account the susceptibility of the human for errors and to design the system to prevent them.
It is said that human factors got its start with aircraft cockpit design in World War II.
The idea of task analysis was first used by human factors specialists in analyzing factory workers' actions on an assembly line.

The passage discusses the origins of computer usability and its connection to related fields, emphasizing the role of human factors and industrial engineering. Here's a breakdown of the key points:

1. **Debate on the Birth of Computer Usability**: The exact birth of computer usability is a matter of debate, suggesting that it evolved gradually over time rather than having a specific starting point.
2. **Influences from Related Fields**: Human-computer interaction (HCI) and usability draw inspiration and knowledge from various related fields. These fields contribute to the development and understanding of usability principles.
3. **Human Factors and System Design**: HCI and usability are closely tied to human behavior and aim to drive the design of systems. Human performance is a critical measure when assessing the effectiveness of these systems.
4. **Addressing Human Error**: The passage highlights a misconception that human factors engineering places blame on humans for errors or system failures. In reality, the focus is on designing systems that consider human susceptibility to errors and prevent them from occurring.
5. **Origins of Human Factors**: Human factors engineering is said to have originated during World War II, particularly in aircraft cockpit design. The concept of task analysis, a key aspect of human factors, was first applied to analyzing the actions of factory workers on assembly lines.

In summary, the passage underscores the interdisciplinary nature of computer usability and human-computer interaction. It acknowledges that usability principles have evolved over time, with influences from fields like human factors engineering and industrial engineering. These fields have contributed valuable insights into designing systems that consider human behavior and aim to prevent errors and failures.

A UX process lifecycle template

Creating a User Experience (UX) process lifecycle template can help streamline and standardize your UX design projects. Here's a template you can use as a starting point. Feel free to adapt it to your specific needs and preferences:

Project Name: [Enter project name]

Project Description: [Provide a brief description of the project]

Team Members:

- [List team members and their roles]

Phase 1: Research and Discovery

1.1 User Research

- Define research objectives
- Conduct user interviews
- Create surveys/questionnaires
- Analyze user feedback and data

1.2 Competitive Analysis

- Identify competitors
- Analyze competitors' products/services
- Document key findings

Phase 2: Analysis and Planning

2.1 Persona Development

- Create user personas
- Define user goals and pain points
- Share personas with the team

2.2 User Journey Mapping

- Map user journeys

- Highlight touchpoints and emotions
- Identify pain points and opportunities

2.3 Information Architecture

- Create IA diagrams
- Define site/app structure
- Plan content organization

Phase 3: Design

3.1 Wireframing

- Develop low-fidelity wireframes
- Review and iterate on wireframes
- Create mid-fidelity wireframes

3.2 Prototyping

- Build interactive prototypes
- Test prototypes with users
- Refine and finalize prototypes

3.3 Visual Design

- Create visual style guide
- Design UI elements
- Apply branding elements

Phase 4: Testing and Validation

4.1 Usability Testing

- Plan usability tests
- Recruit participants
- Conduct usability tests
- Analyze test results

4.2 A/B Testing

- Define A/B test variations
- Implement A/B tests
- Monitor and analyze test data
- Make data-driven decisions

4.3 Iterative Design

- Gather feedback from stakeholders
- Make design improvements
- Iterate on design based on user feedback

Phase 5: Implementation and Development

5.1 Collaboration

- Collaborate with development team
- Ensure design implementation matches the approved design

5.2 Quality Assurance

- Conduct QA testing
- Identify and report bugs
- Ensure design fidelity

Phase 6: Launch and Deployment

6.1 Release

- Plan release date
- Deploy the product or feature
- Monitor the launch for issues

Phase 7: Post-Launch Evaluation

7.1 User Feedback

- Gather user feedback
- Monitor user reviews and comments
- Address user-reported issues

7.2 Analytics and Metrics

- Monitor key performance indicators (KPIs)
- Analyze user behavior data
- Identify areas for improvement

Phase 8: Maintenance and Iteration

8.1 Regular Updates

- Schedule regular maintenance
- Address bug fixes and minor improvements
- Keep the product up to date

8.2 Major Updates

- Plan for major updates or redesigns
- Conduct user research for major changes
- Iterate on the design

Phase 9: Scaling and Growth

9.1 Scaling Strategy

- Explore strategies for scaling
- Consider expansion to new markets
- Add new features or functionalities

Phase 10: End of Life

10.1 Retirement Planning

- Assess the product's lifecycle
- Decide on the retirement strategy (if applicable)
- Communicate the product's end of life to users

Project Conclusion and Documentation:

- Summarize key learnings and successes

- Document any unresolved issues or lessons for future projects

This template provides a structured framework for managing a UX design project from start to finish. You can customize it by adding specific project details, timelines, and assigning responsibilities to team members. Additionally, you can use project management tools or software to track progress and collaborate with your team effectively.

Choosing a process instance for your project

Selecting the right process instance for your project involves considering both project parameters (inputs) and process parameters (outputs). Here's a breakdown of the factors and considerations related to each:

Project Parameters (Inputs to Process Choices):

1. **Project Goals:**
    - Consider the specific objectives and outcomes you want to achieve with the project. Are you aiming for a rapid product launch, user satisfaction, or long-term innovation?
2. **Risk Tolerance:**
    - Assess how tolerant your team and stakeholders are to project risks. Some processes provide more predictability and control, while others embrace uncertainty and change.
3. **Project Resources:**
    - Evaluate the availability of resources, including budget, time, and skilled team members. Some processes may require more extensive resources.
4. **Type of System Being Designed:**
    - The complexity and nature of the system you are designing can influence your process choice. Highly complex systems may benefit from iterative and adaptive methodologies.
5. **Development Organizational Culture:**

- Consider the existing culture and practices within your organization. Are there established development methodologies, and are teams open to change?

6. **Stage of Progress Within Project:**
    - The current stage of your project can impact your process choice. Early stages may focus on research and ideation, while later stages may require a more structured development approach.

Process Parameters (Outputs of Process Choices):

1. **Rigorous UX Processes vs. Rapid Methods:**
    - Decide on the level of rigor and formality you want in your UX process. Some projects may benefit from a highly structured and detailed approach, while others may require quick iterations and rapid development.
2. **Data Collection Techniques:**
    - Choose appropriate methods for gathering user feedback and insights. This can include user interviews, surveys, usability testing, analytics, and more.
3. **Agile UX Process:**
    - Consider whether an agile UX process aligns with your project's needs. Agile methodologies emphasize flexibility, collaboration, and responsiveness to change. You can adopt an agile approach for the entire project or specific phases.

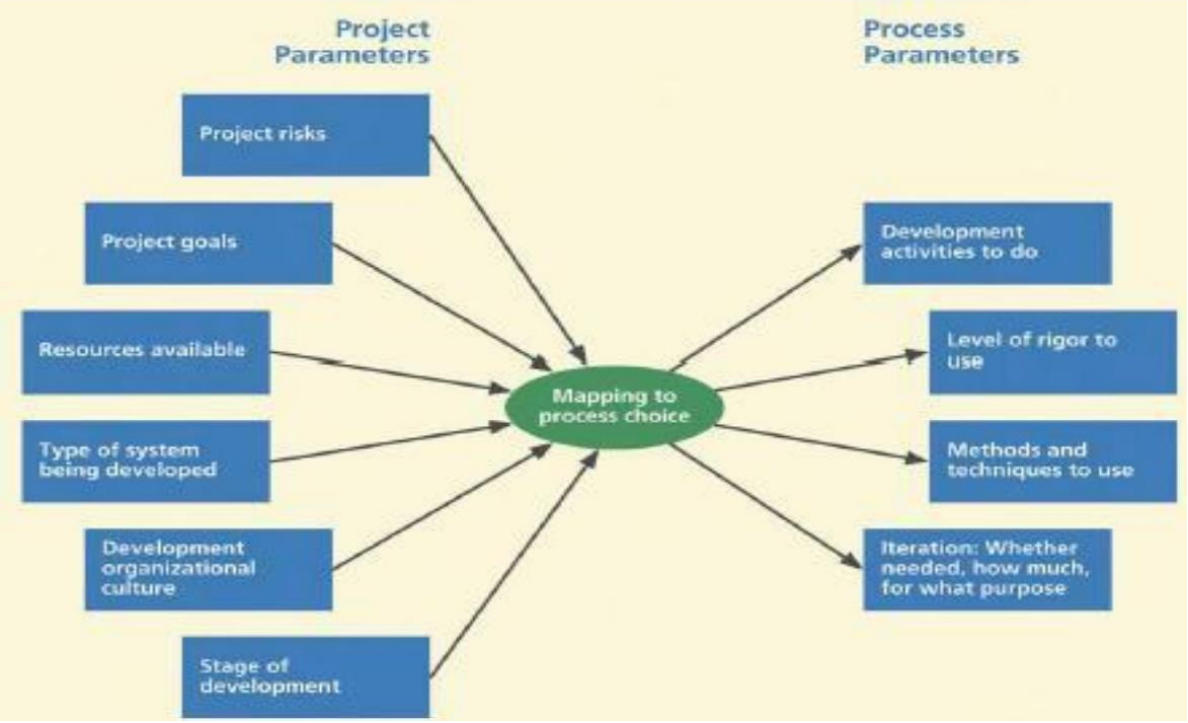
Incorporate these factors into your decision-making process to choose the most suitable approach. Here are a few examples of how these considerations might influence your choice:

- **Highly Innovative Project with Limited Resources:** In this case, you might opt for a Lean Startup approach, which allows you to quickly validate ideas with minimal resources.
- **Complex Software Development Project with Well-Defined Requirements:** You could choose a Waterfall or a traditional UX process with a focus on detailed documentation and structured phases.

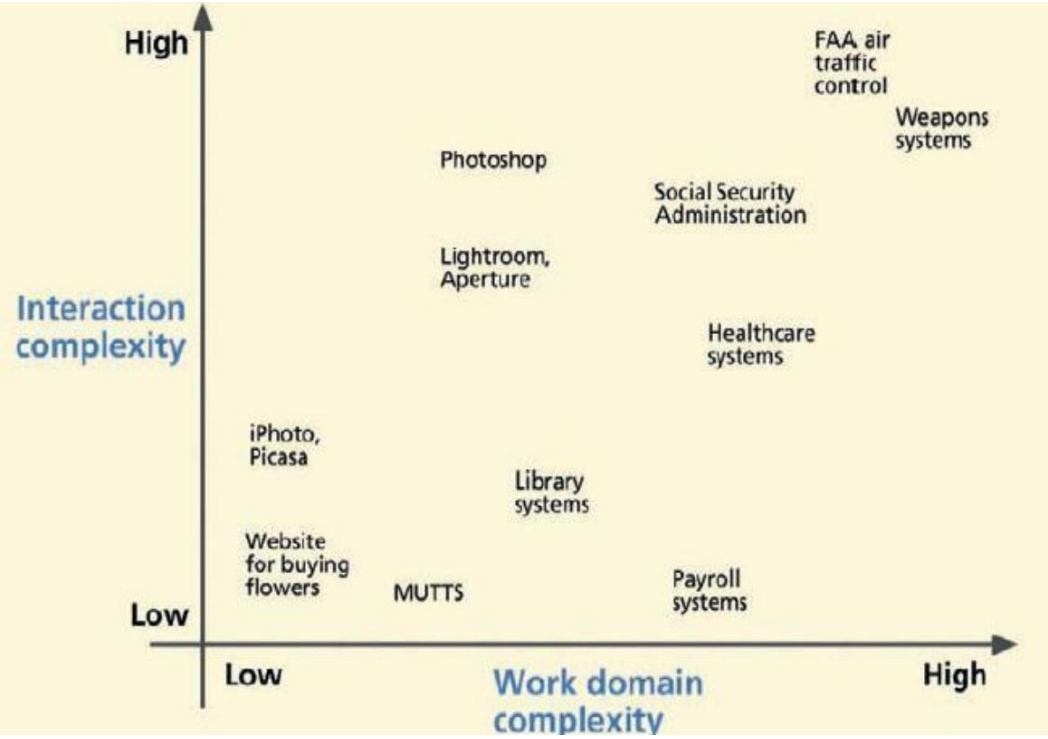
- **Evolving Requirements and Agile Culture:** An Agile UX process such as Scrum or Kanban could be a good fit for a project that requires continuous adaptation and collaboration.
- **Mixed Methods for Comprehensive Insights:** If your project requires a holistic understanding of user needs, combining various data collection techniques, including user interviews, surveys, and usability testing, may be necessary.

Ultimately, the choice of a process instance should align with your project's unique characteristics and goals, striking a balance between structure and adaptability based on your project parameters and process parameters.

• Mapping Project Parameters to Process Choices



The system complexity space



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

  Low interaction complexity

  High interaction complexity

- Work domain complexity - represented on the horizontal axis which is about the degree of intricacy and the technical nature of the corresponding field of work

  Low Work Domain Complexity

  High Work Domain Complexity

Interaction complexity and work domain complexity are two key concepts in the context of designing and understanding systems, particularly in the field of

human-computer interaction (HCI) and systems engineering. Let's explore the meaning of these terms:

1. **Interaction Complexity:**

Interaction complexity refers to the intricacy or elaborateness of user actions and the level of cognitive effort required to accomplish tasks when using a system, software, device, or interface. It involves evaluating how users interact with a system and the challenges they encounter during the interaction process. Key aspects of interaction complexity include:

- **Number of Steps:** The number of actions or steps users need to perform to achieve their goals within the system.
- **Cognitive Load:** The mental effort required by users to understand and navigate the system, make decisions, and complete tasks.
- **User Interface Design:** The design of user interfaces, including the layout, navigation, and visual elements, which can either simplify or complicate interactions.
- **Feedback and Error Handling:** The clarity and effectiveness of feedback provided to users during interactions, as well as the system's ability to handle errors gracefully.

Interaction complexity can vary widely, from simple and intuitive interactions in user-friendly applications to complex and cognitively demanding interactions in specialized software or systems.

2. **Work Domain Complexity:**

Work domain complexity refers to the degree of intricacy and the technical nature of the field or domain in which a system operates or supports. It assesses the complexity of the tasks, processes, and activities that the system is designed to facilitate or address. Key aspects of work domain complexity include:

- **Specialized Knowledge:** The level of specialized knowledge, expertise, or domain-specific skills required to effectively use or work within the domain.
- **Technical Challenges:** The complexity of the technical processes, calculations, or operations involved in the domain.

- **Safety and Criticality:** The degree to which errors or failures within the domain can have serious consequences, such as in medical, aerospace, or financial domains.
- **Interconnectedness:** The extent to which the domain interacts with other systems, processes, or external factors, leading to complex dependencies.

Work domain complexity can range from low complexity in familiar and straightforward domains (e.g., basic e-commerce) to high complexity in specialized and critical domains (e.g., healthcare diagnosis or aerospace engineering).

Understanding both interaction complexity and work domain complexity is crucial for designing user-friendly systems and interfaces that align with users' cognitive capabilities and the demands of the specific work domain. Balancing these complexities is essential to optimize user experiences and system performance while ensuring safety and effectiveness in complex domains.

The two-dimensional complexity space you've described, with interaction complexity on the vertical axis and work domain complexity on the horizontal axis, is a valuable framework for understanding and categorizing systems based on these two critical dimensions. Let's delve deeper into each quadrant of this complexity space:

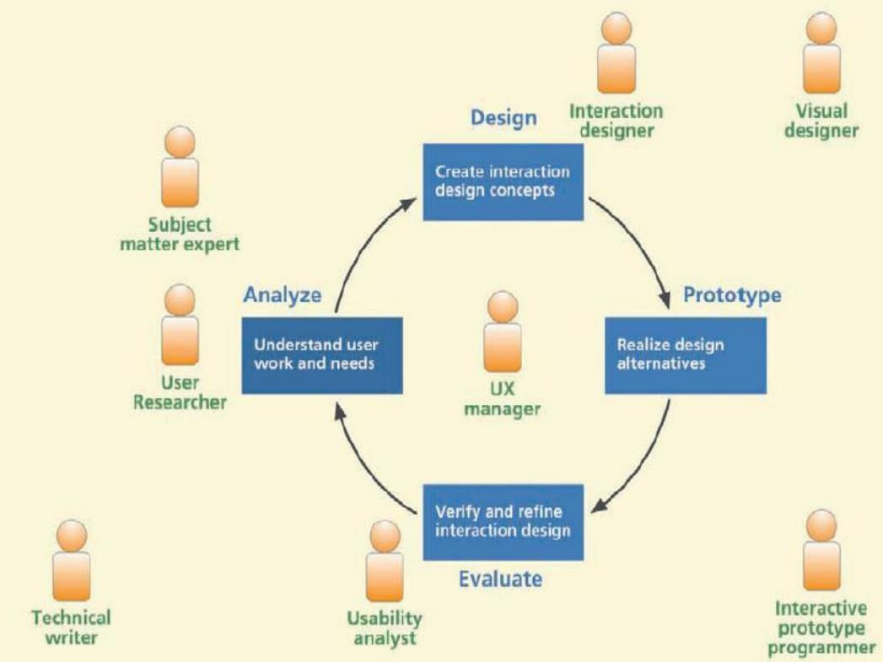
1. **Low Interaction Complexity, Low Work Domain Complexity:**
    - This quadrant represents systems where both user interactions and the work domain are relatively simple.
    - Examples could include basic consumer apps, straightforward websites, or common household appliances.
    - For such systems, simple and agile design and development processes may be appropriate. The emphasis may be on ease of use, quick development cycles, and iterative improvements.
2. **High Interaction Complexity, Low Work Domain Complexity:**
    - Here, user interactions are complex, but the underlying work domain is relatively straightforward.
    - Examples might include complex video games, interactive simulations, or creative software tools.

- In these cases, the design process may prioritize user experience, interaction design, and engagement strategies. Agile methods may also be useful to iterate on the user experience.

3. **Low Interaction Complexity, High Work Domain Complexity:**
    - In this quadrant, user interactions are simple, but the work domain is highly complex and technical.
    - Examples could involve scientific research tools, complex industrial systems, or specialized software for experts.
    - Design processes may focus on understanding and modeling the intricate work domain while keeping user interactions as straightforward as possible. Systems engineering and domain expertise may play a significant role.
4. **High Interaction Complexity, High Work Domain Complexity:**
    - This quadrant represents systems where both user interactions and the work domain are complex.
    - Examples might include advanced medical equipment, sophisticated financial systems, or cutting-edge research tools.
    - For such complex systems, comprehensive and rigorous design processes are often necessary. This may involve detailed user research, in-depth system modeling, and iterative design approaches.

Understanding where a system falls within this complexity space is essential for making informed decisions about the appropriate design, development, and management processes. The chosen methodologies, tools, and approaches should align with the specific challenges posed by the system's interaction complexity and work domain complexity. Tailoring the process to the characteristics of the system can lead to more effective and successful outcomes.

Meet the user interface team



1. **User Researcher:**
    - Job: Talks to users and observes them to understand how they use a product.
    - Goal: To figure out what users like and dislike about a product and what problems they face.
2. **Users, User Representatives, Customers, and Experts:**

- Job: These are the people who use the product or know a lot about it.
- Goal: Share their thoughts and experiences to help make the product better.

3. **User Interaction Designer:**
    - Job: Draws and designs how the product looks and how users can use it.
    - Goal: To make the product easy to use and good-looking.
4. **UX Analyst or Evaluator:**
    - Job: Checks if the product is easy to use and if there are any problems.
    - Goal: Find and fix any issues to make the product work better for users.
5. **Technical Writer:**
    - Job: Writes instructions and explanations inside the product.
    - Goal: Makes sure users can understand and use the product easily.
6. **Visual/Graphic Designer:**
    - Job: Makes the product look nice with colors, pictures, and logos.
    - Goal: Gives the product a professional and attractive appearance.
7. **Interactive Prototype Programmer:**
    - Job: Builds a pretend version of the product that you can click on and use.
    - Goal: To show how the real product will work and get feedback.
8. **UX Manager:**
    - Job: Leads the team and makes sure everyone is doing their job.
    - Goal: To make sure the whole design process goes smoothly and the product is user-friendly.

Having all these roles on the team helps create a product that looks good, works well, and is easy for people to use. Team members work together to understand users, design the product, and make it the best it can be.

Scope of UX presence within the team

The scope of UX (User Experience) presence within a team can vary depending on the organization, project, and the team's specific needs and goals. However, in a typical team setting, the scope of UX presence encompasses various aspects and stages of a project:

1. **Early Planning and Strategy:**
    - UX researchers and strategists may be involved in the early stages to identify user needs, market trends, and competitive analysis. They help define project goals and align them with user-centric objectives.
2. **User Research and Analysis:**
    - UX researchers play a crucial role in understanding user behaviors, preferences, and pain points. They conduct user interviews, surveys, and observations to gather insights that inform design decisions.
3. **Conceptual Design and Ideation:**
    - User interaction designers participate in brainstorming and idea generation sessions. They help create the initial concepts and sketches for the product's user interface.
4. **Prototyping and Wireframing:**
    - UX designers translate concepts into tangible prototypes and wireframes. These low-fidelity representations help visualize the design and gather feedback early in the process.
5. **User Testing and Feedback:**
    - UX professionals plan and conduct usability testing sessions with real users. They collect feedback on the product's usability and iterate on the design based on user input.
6. **Visual and Graphic Design:**

- Visual and graphic designers work on the look and feel of the product. They ensure that the interface is visually appealing and aligns with branding guidelines.

7. **Documentation and Content Strategy:**
    - Technical writers and content strategists focus on creating user-friendly documentation and content within the product. They ensure that users can easily understand and use the system.
8. **Interaction Design and User Interface (UI) Design:**
    - UX designers are heavily involved in creating the detailed design of the user interface, including layout, buttons, forms, and navigation elements. They aim to make the interface intuitive and user-friendly.
9. **Accessibility and Inclusivity:**
    - Accessibility experts work to ensure that the product is usable by individuals with disabilities. They focus on compliance with accessibility standards and guidelines.

More about UX lifecycles

It seems like you've provided excerpts or sections from a document discussing the need for a broader user experience (UX) process, fundamental activities in building something (whether a product or software), parallel streams of software and interaction process activities, the lightweight nature of interaction design iteration, and the pre-design part of the UX lifecycle. Let's break down these concepts:

1. Broadening the UX Process:

- The passage emphasizes that while usability testing is valuable, it should not be the sole focus of the UX process. Instead, it advocates for incorporating other User-Centered Design (UCD) activities earlier in the UX process. This broader approach ensures better product quality.

2. Fundamental Activities in Building Anything:

- Building anything, whether it's a physical structure like a house or a digital product like software, involves two fundamental activities: design

and implementation. These activities are essential building blocks for creating something.

3. Managing Complexity:

- As the complexity of the target system or product increases, additional steps in the development process become necessary to effectively manage that complexity. Complexity can require more comprehensive planning and execution.

4. Parallel Streams of Software and Interaction Process Activities:

- This section likely discusses the distinction between software design and software implementation in the software development process. It may also mention how these activities can run in parallel but have different focuses and goals.

5. Iteration for Interaction Design Refinement:

- It suggests that the process of refining interaction design can be lightweight and efficient. Iteration in this context is cost-effective, especially when done early in the development lifecycle. It minimally impacts schedules and can be carried out alongside other development activities.

6. The Pre-Design Part of the UX Lifecycle:

- This section appears to introduce the pre-design phase of the UX lifecycle. It may discuss activities such as contextual inquiry (studying users in their natural environments), context extraction (gathering relevant information), analysis, identifying needs and requirements, and modeling. These activities lay the groundwork for the design phase of the UX process.

Overall, the document seems to emphasize the importance of a comprehensive UX process that extends beyond usability testing, and it highlights the role of early-stage activities in shaping the design and development of products or software. The use of lightweight iterations and parallel streams of activities can help manage complexity and improve the overall quality of the final product.

Refer ppt

The system concept statement

- A system concept statement is a concise descriptive summary of the envisioned system or product stating an initial system vision or mandate; in short, it is a mission statement for the project.

- A system (or product) concept statement is where it all starts, even before contextual inquiry.

- Before a UX team can conduct contextual inquiry, which will lead to requirements and design for the envisioned system, there has to be a system concept.

Points To Remember :

- A system concept statement is typically 100 to 150 words in length.
- It is a mission statement for a system to explain it to outsiders and to help set focus and scope for system development internally.
- Writing a good system concept statement is not easy.
- The amount of attention given per word is high. A system concept statement is not just written; it is iterated and refined to make it as clear and specific as possible.

An effective 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? (You need to be broad here to include business objectives.)
- What is the design vision and what are the emotional impact goals? In other words, what experience will the system provide to the user? This factor is especially important if the system is a commercial product.

Example: System Concept Statement for the Ticket Kiosk System

Here is an example of a system concept statement that we wrote for the 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 rapidly purchase tickets 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. (139 words)

Upon interacting with the customers and users, some of our objectives in this system concept statement will be adjusted and assumptions will be corrected.

A system concept statement is a concise and clear description of the fundamental idea or concept behind a system, project, or initiative. It serves as

a high-level overview of what the system is intended to achieve, its purpose, and its key features. A well-crafted system concept statement helps stakeholders, including project teams and decision-makers, understand the project's scope and objectives. Here's what a system concept statement typically includes:

1. **System Name:** The name or title of the system, project, or initiative being described.
2. **Purpose/Objective:** A brief statement of the primary goal or purpose of the system. What problem or need does it address? What is it intended to achieve or improve?
3. **Scope:** An outline of the boundaries or limits of the system. This may include what is included in the system and what is excluded. It helps define the project's focus.
4. **Key Features/Functionality:** A list of the essential features, functions, or capabilities that the system will offer. This provides an overview of what users can expect from the system.
5. **Target Audience/Users:** A description of the intended users or audience for the system. Who will benefit from or interact with the system?
6. **Value Proposition:** A statement of the value or benefits that the system provides to its users or stakeholders. What problem does it solve, and what advantages does it offer?
7. **Unique Selling Points (USPs):** Any distinctive qualities or aspects of the system that set it apart from similar solutions or competitors.
8. **High-Level Requirements:** A summary of the major requirements or constraints that the system must adhere to. This can include technical, budgetary, or timeline constraints.
9. **Dependencies:** Any external factors, systems, or components that the project depends on or that depend on the project.
10. **Assumptions and Risks:** A list of assumptions made during the planning phase and potential risks or uncertainties associated with the project.
11. **Strategic Alignment:** How the project aligns with the organization's strategic goals or objectives.

12. **Project Team:** A brief overview of the team or stakeholders responsible for the project, including roles and responsibilities.
13. **Timeline:** A high-level estimate of the project's timeline, including key milestones or phases.

A well-crafted system concept statement provides a clear and shared understanding of the project's purpose and scope, helping to guide decision-making, secure buy-in from stakeholders, and serve as a reference point throughout the project's lifecycle. It's often one of the initial documents produced during project planning and can evolve as the project progresses and more detailed information becomes available.

Certainly, here's a shorter example of a system concept statement for a mobile app called "HealthTrack":

System Concept Statement: HealthTrack

1. System Name: HealthTrack

2. Purpose/Objective: HealthTrack is a mobile app designed to help users monitor their daily health and fitness activities, track their progress over time, and make informed decisions to improve their overall well-being.

3. Scope: The app includes features for tracking steps, calories burned, sleep patterns, and nutrition. It does not provide medical advice or diagnostics.

4. Key Features/Functionality:

- Daily activity tracking with a user-friendly interface.
- Integration with wearable fitness devices for real-time data sync.
- Personalized health insights and goal setting.
- Secure data storage compliant with privacy regulations.

5. Target Audience/Users: Health-conscious individuals interested in maintaining an active lifestyle and making healthier choices.

6. Value Proposition: HealthTrack empowers users to take control of their health by providing easy-to-understand data and personalized recommendations, leading to improved fitness and well-being.

7. Unique Selling Points (USPs):

- Seamless integration with popular fitness wearables.
- Data-driven insights for actionable health decisions.
- User-friendly and intuitive interface.

8. High-Level Requirements: The app must be available on iOS and Android platforms, support data synchronization with wearables, and ensure data security.

9. Dependencies: HealthTrack depends on access to smartphone hardware for tracking, and it requires user consent for data collection.

10. Assumptions and Risks: Assumption: Users have compatible wearable devices. Risks: Data privacy concerns and potential technical issues with data synchronization.

11. Strategic Alignment: HealthTrack aligns with our organization's mission to promote healthier lifestyles through technology.

12. Project Team: The project team consists of mobile app developers, UX designers, and data analysts.

13. Timeline: The project is estimated to take 9 months, with milestones including app development, testing, and a public launch.

User work activity gathering

To do your user work activity data gathering you will:

- Prepare and conduct field visits to the customer/user work environment, where the system being designed will be used.
- Observe and interview users while they work.
- Inquire into the structure of the users' own work practice
- Learn about how people do the work your system is to be designed to support
- Take copious, detailed notes, raw user work activity data, on the observations and interviews

---

3.1 Before the Visit: Preparation for the Domain-Complex System Perspective

- Learn about your customer organization before the visit
- Learn about the domain
- Issues about your team
- Lining up the right customer and user people
- Get access to "key" people
- What if you cannot find real users?
- Setting up the right conditions
- How many interviewees at a time?
- Preparing your initial questions
- Before the visit: Preparation for the product perspective
- Anticipating modeling needs in contextual inquiry: Create contextual data "bins"

7) Data Bin - a temporary repository—for example, a labeled pile of notes on a table—to hold data—raw contextual data at first and, later, synthesized work activity notes. Each bin corresponds to a different data category or contextual data topic.

3.2 During the Visit: Collecting User Work in the Domain-Complex System Perspective

Activity	Data
☐ When you first arrive	
☐ Remember the goal	
☐ Establish trust and rapport	
☐ Form partnerships with users	
☐ Task data from observation and interview	
☐ Recording video	
☐ Note taking	
☐ Use a numbering system to identify each point in data	

- How to proceed
  - Be a listener; in most cases you should not offer your opinions about what users might need.
  - Do not lead the user or introduce your own perspectives.
  - Do not expect every user to have the same view of the work domain and the work; ask questions about the differences and find ways to combine to get the "truth."
  - Capture the details as they occur; do not wait and try to remember it later.
  - Be an effective data detective. Follow leads and discover, extract, "tease out" and collect "clues." Be ready to adapt, modify, explore, and branch out.

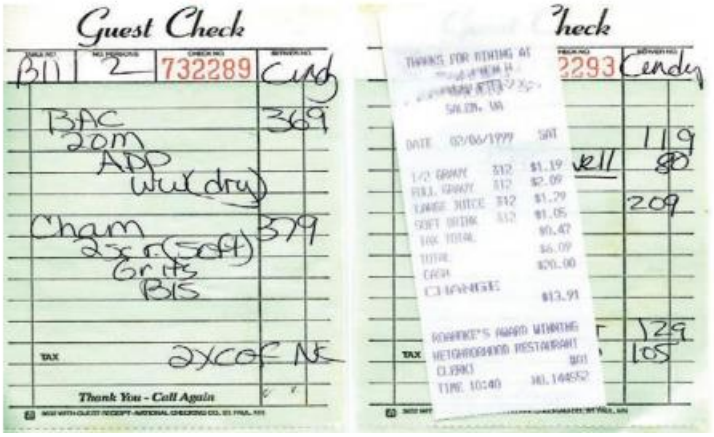
- Pay attention to information needs of users
- What about design ideas that crop up?
- What about analyst and designer ideas that crop up?
- Questions not to ask
  - Do not ask about the future
  - Do not ask for design advice, how they would design a given feature
  - Do not ask a question by trying to state what you think is their rationale.

Other forms of data collection

- Copious digital pictures of the physical environment, devices, people at work, and anything else to convey work activities and context visually
- On-the-fly diagrams of workflow, roles, and relationships
- On-the-fly sketches of the physical layout, floor plans (not necessary to be to scale), locations of people, furniture, equipment, communications connections, etc.
- Quantitative data—for example, how many people do this job, how long do they typically work before getting a break, or how many widgets per hour do they assemble on the average?

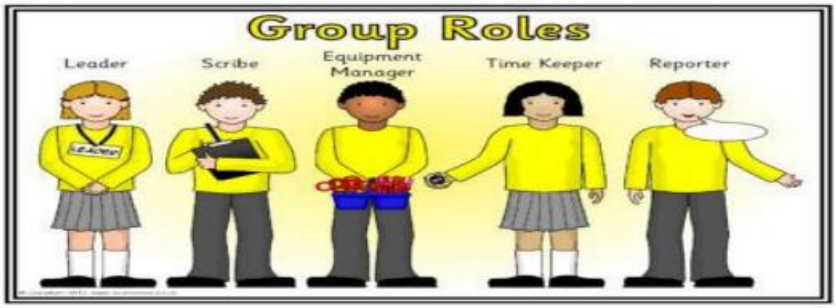
Collect work artifacts

- a set of paper work artifacts, including manually created order forms and "guest checks," from a local restaurant



3.3 During the Visit: Collecting User Work Activity Data in the Product Perspective

- Roles of users will be different with commercial products.
- In most cases, work in a domain-complex system context is performed by people in roles that make up the organization, which we will be calling "**work roles**"



Example: User Data Gathering for MUTTS

- There is contextual inquiry sessions, interviewing MUTTS employees and customers.
- Where three analysts separately interviewing several groups of one or two users at a time and came up with a fairly rich set of raw data transcripts.

Q: We want to begin with some questions about your usage of the ticket service, MUTTS. What do you do for a living? Tell us about your typical day.

A: I have a 9 to 5 job as a lab technician in Smyth Hall. However, I often have to work later than 5PM to get the job done.

Q: So do you use MUTTS to buy tickets for entertainment?

A: I work long hours and, at the end of the day, I usually do not have the energy to go to MUTTS for entertainment tickets. Because this is the only MUTTS location, I cannot buy tickets during normal working hours, but the MUTTS window is not open after 7PM.

Look for emotional aspects of work practice

Points To Remember:

- Look for the impact of aesthetics and fun in work practice, and look for opportunities for more of the same.
- You must try harder to uncover an understanding about emotional and social aspects of work practice.
- Look for ways to fight job boredom

  - What about the work is boring?
  - What are they doing when they have fun?
  - Where can job stress be relived with aesthetics & fun?
  - What parts of usage are learned over longer times?

Understanding the emotional aspects of work practices is a crucial aspect of user-centered design (UCD) and human-centered design. Emotions can significantly influence how individuals perform tasks, interact with systems, and experience their work environments. Here are some ways to look for emotional aspects in work practices:

1. **User Interviews:** During user interviews, ask participants about their emotional experiences related to their work. For example:
   - "How do you feel when you encounter challenges in your daily tasks?"
   - "Can you describe any moments of frustration or satisfaction in your work?"
   - "What emotions are triggered when using the current system or tools?"
2. **Observation and Contextual Inquiry:** Observe users in their work environment to identify emotional cues. Pay attention to facial expressions, body language, and verbal expressions of frustration, joy, stress, or satisfaction.
3. **Surveys and Questionnaires:** Include questions in surveys or questionnaires that directly address emotional aspects of work. For example:

- "On a scale of 1 to 5, how would you rate your level of frustration when using the current system?"
- "What aspects of your work bring you the most joy or satisfaction?"

4. **Emotion Tracking:** Implement methods to track and record emotional states throughout the workday. This can be done through self-reporting by users or by using wearable devices that monitor physiological indicators like heart rate or skin conductance.
5. **Diary Studies:** Encourage participants to maintain work diaries in which they record their emotions and experiences at various points during their workday. This provides a longitudinal view of emotional fluctuations.
6. **Empathy Mapping:** Create empathy maps that visualize the emotions, thoughts, and behaviors of users. This helps design teams better understand the emotional aspects of user experiences.
7. **User Stories:** Use user stories to capture the emotional context of tasks. For example, a user story might begin with, "As a user, I feel frustrated when..."
8. **Affinity Diagramming:** Organize observations and findings into affinity diagrams, which can help identify common emotional themes and pain points among users.
9. **Persona Development:** Incorporate emotional characteristics and motivations into user personas. This makes it easier to empathize with users and design solutions that resonate with their emotions.
10. **Workshop and Co-creation Sessions:** Conduct workshops or co-creation sessions with users to collaboratively explore emotional aspects of work practices. Encourage participants to share their feelings and experiences openly.
11. **Usability Testing with Emotional Metrics:** When conducting usability testing, consider adding emotional metrics to evaluate participants' emotional responses as they interact with a prototype or system.

By actively seeking out and addressing emotional aspects of work practices, designers and researchers can create more empathetic and user-friendly

solutions. Understanding how users feel while performing tasks can lead to designs that not only meet functional needs but also enhance overall satisfaction and well-being in the workplace.