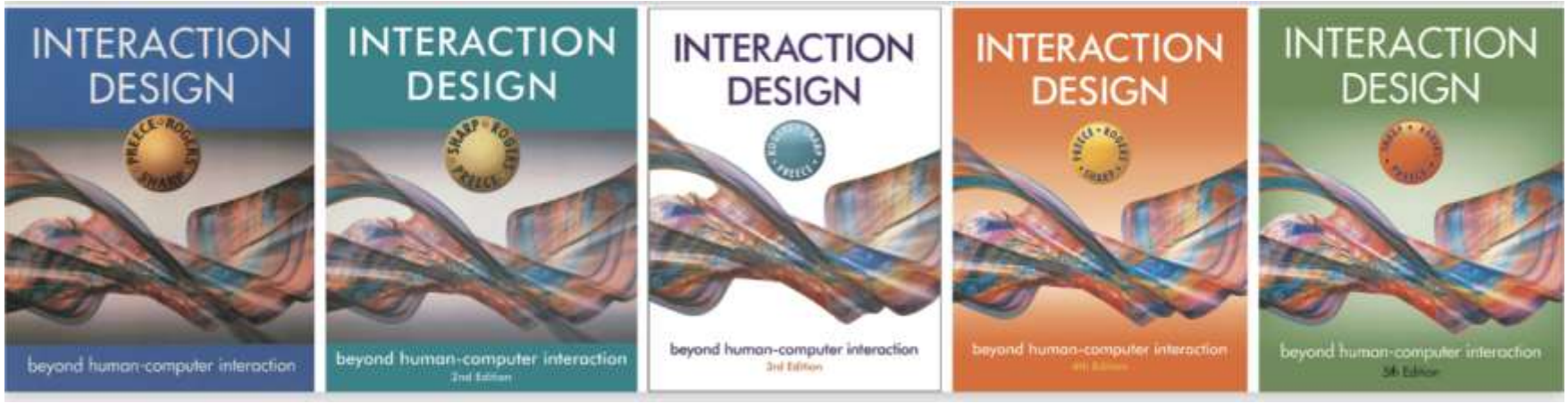
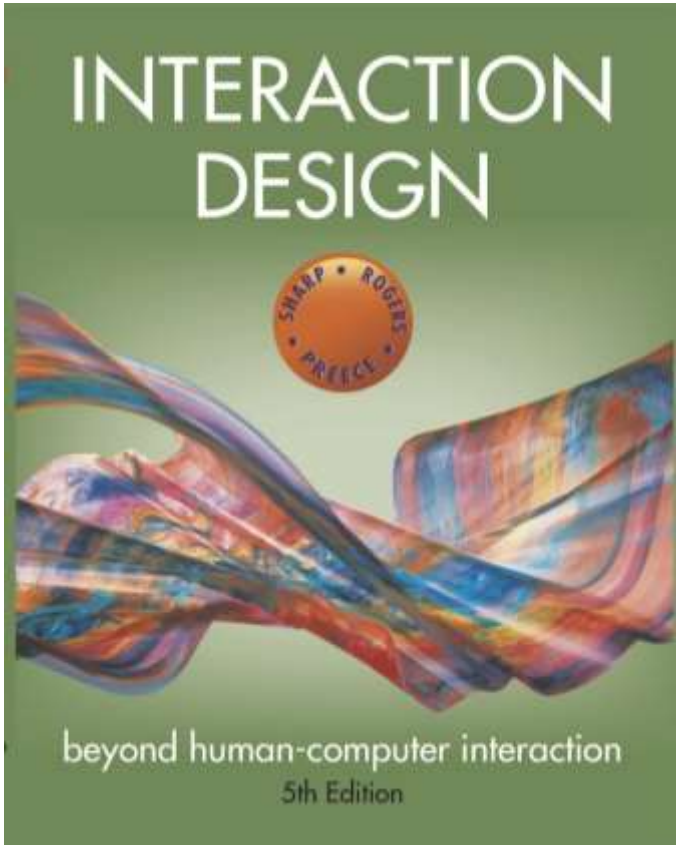


Helen Sharp, Yvonne Rogers, and Jenny Preece (2019)



Chapter 1

WHAT IS INTERACTION DESIGN?

Learning Objectives

- **The main goals of this chapter are to accomplish the following:**
  - Explain the difference between good and poor interaction design.
  - Describe what interaction design is and how it relates to human-computer interaction and other fields.
  - Explain the relationship between the user experience and usability.
  - Introduce what is meant by accessibility and inclusiveness in relation to human computer interaction.
  - Describe what and who is involved in the process of interaction design.
  - Outline the different forms of guidance used in interaction design.
  - Enable you to evaluate an interactive product and explain what is good and bad about it in terms of the goals and core principles of interaction design.

Bad designs

Elevator controls and labels on the bottom row all look the same, so it is easy to push a label by mistake instead of a control button.

www.baddesigns.com

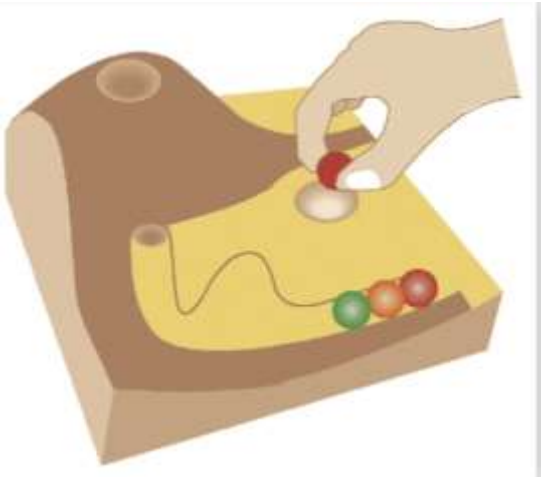
People do not make same mistake for the labels and buttons on the top row. Why not?

Why is this vending machine so bad?

www.baddesigns.com

- Need to push button first to activate reader
- Normally insert bill first before making selection
- Contravenes well known convention

Good design



- Marble answering machine (Bishop, 1995)
- Based on how everyday objects behave
- Easy, intuitive, and a pleasure to use
- Only requires one-step actions to perform core tasks

Good and bad design

Why is the TiVo remote much better designed than standard remote controls?

- Peanut shaped to fit in hand
- Logical layout and color-coded, distinctive buttons
- Easy-to-locate buttons

Dilemma

Which is the best way to interact with a smart TV? Why?

- Pecking using a grid keyboard via a remote control
- Swiping across two alphanumeric rows using a touchpad on a remote control
- Voice control using remote or smart speaker

What to design

Need to take into account:

- Who the users are
- What activities are being carried out
- Where interaction is taking place

Need to optimize the interactions users have with a product:

- So that they match the users' activities and needs

What is interaction design?

"Designing interactive products to support the way people communicate and interact in their everyday and working lives."

Sharp, Rogers, and Preece (2019)

"The design of spaces for human communication and interaction."

Winograd (1997)

Goals of interaction design

Develop usable products

- Usability means easy to learn, effective to use, and provides an enjoyable experience

Involve users in the design process

Which kind of design?

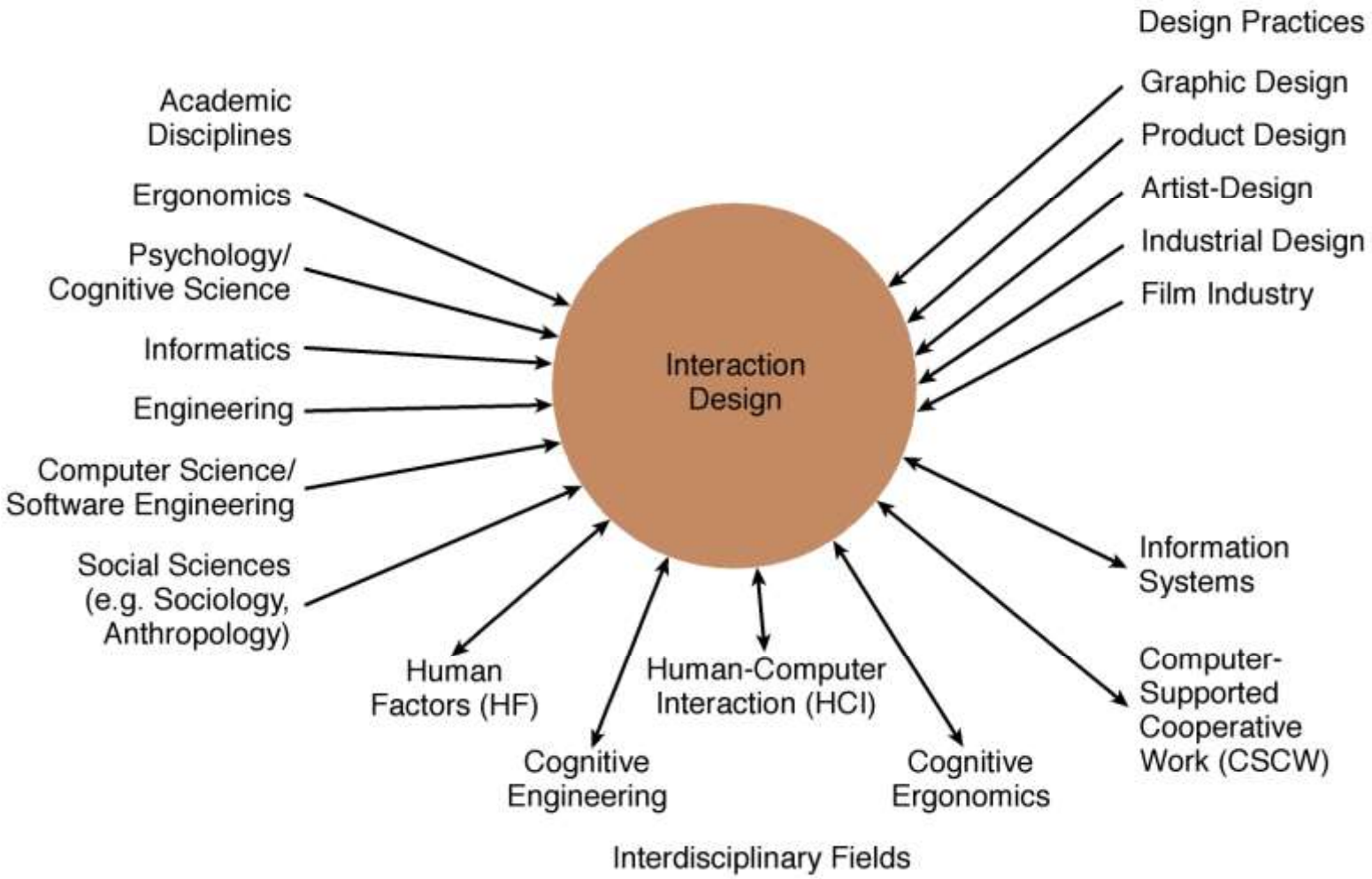
Number of other terms used emphasizing what is being designed, for example:

- User interface design, software design, user-centered design, product design, web design, experience design (UX)

Interaction design is the umbrella term covering all of these aspects:

- Fundamental to all disciplines, fields, and approaches concerned with researching and designing computer-based systems for people

Interaction design



Relationship between ID, HCI, and other fields−academic disciplines

Academic disciplines contributing to ID:

- Psychology
- Social Sciences
- Computing Sciences
- Engineering
- Ergonomics
- Informatics

Relationship between ID, HCI and other fields−design practices

Design practices contributing to ID:

- Graphic design
- Product design
- Artist-design
- Industrial design
- Film industry

Relationship between ID, HCI and other fields−interdisciplinary fields

Interdisciplinary fields that ‘do’ interaction design:

- HCI
- Ubiquitous Computing
- Human Factors
- Cognitive Engineering
- Cognitive Ergonomics
- Computer Supported Co-operative Work
- Information Systems

Working in multidisciplinary teams

- Many people from different backgrounds involved
- Different perspectives and ways of seeing and talking about things

Benefits

- More ideas and designs generated

Disadvantages

- Difficult to communicate and progress forward the designs being create

Interaction design in business

Large number of ID consultancies. Examples of well known ones include:

- **Nielsen Norman Group**: “help companies enter the age of the consumer, designing human-centered products and services”
- **Cooper:** “From research and product to goal-related design”
- **IDEO: “**creates products, services and environments for companies pioneering new ways to provide value to their customers”

The user experience

How a product behaves and is used by people in the real world

- The way people feel about it and their pleasure and satisfaction when using it, looking at it, holding it, and opening or closing it
- “Every product that is used by someone has a user experience: newspapers, ketchup bottles, reclining armchairs, cardigan sweaters.” (Garrett, 2010)
- “All aspects of the end-user's interaction with the company, its services, and its products. (Nielsen and Norman, 2014)

Cannot design a user experience−only can design *for* a user experience

Defining user experience

How users perceive a product, such as whether a smartwatch is seen as sleek or chunky, and their emotional reaction to it, such as whether people have a positive experience when using it.

(Hornbæk and Hertzum, 2017)

Hassenzahl's (2010) model of the user experience

- Pragmatic: how simple, practical, and obvious it is for the user to achieve their goals
- Hedonic: how evocative and stimulating the interaction is to users

Why was the iPod user experience such a success?

Figure 1.6 The iPod Nano Touch

Source: ©Press Association, reproduced with permission.

- Quality user experience from the start
- Simple, elegant, distinct brand, pleasurable, must have fashion item, catchy names, cool...

Core characteristics of interaction design

- Users should be involved throughout the development of the project
- Specific usability and user experience goals need to be identified, clearly documented, and agreed to at the beginning of the project
- Iteration is needed through the core activities

Why?

Help designers:

- Understand how to design interactive products that fit with what people want, need, and may desire
- Appreciate that one size does not fit all (for example, teenagers are very different to grown-ups)
- Identify any incorrect assumptions they may have about particular user groups. (for example, not all old people want or need big fonts)
- Be aware of both people's sensitivities and their capabilities

Accessibility and inclusiveness

Accessibility*:* the extent to which an interactive product is accessible by as many people as possible

- Focus is on people with disabilities; for instance, those using android OS or apple voiceover

Inclusiveness: making products and services that accommodate the widest possible number of people

- For example, smartphones designed for all and made available to everyone regardless of their disability, education, age, or income

Disabilities

- Whether someone is disabled changes over time with age, or recovery from an accident
- The severity and impact of an impairment can vary over the course of a day or in different environmental conditions
- Disabilities can result because technologies are designed to necessitate a certain type of interaction that is impossible for someone with an impairment

Understanding disability

Disabilities can be classified as:

- Sensory impairment (such as loss of vision or hearing)
- Physical impairment (having loss of functions to one or more parts of the body after a stroke or spinal cord injury)
- Cognitive (including learning impairment or loss of memory/cognitive function due to old age)

Each type can be further defined in terms of capability:

- For example, someone might have only peripheral vision, be color blind, or have no light perception

Impairment can be categorized:

- Permanent (for instance, long-term wheelchair user)
- Temporary (that is, after an accident or illness)
- Situational (for example, a noisy environment means that a person can’t hear)

Being cool about disability

- Prosthetics can be designed to move beyond being functional (and often ugly) to being desirable and fashionable
- People now refer to “wearing their wheels,” rather than “using a wheelchair”

Fashionable leg cover designed by Alleles Design Studio

Cultural differences

5/6/2015 versus 21/5/2015?

- Which should be used for international services and online forms?
- Why is it that certain products, like smartphones, are universally accepted by people from all parts of the world, whereas people from different cultures react to websites differently?

Usability and user experience goals

- Part of the process of understanding users is to be clear about the primary objective of developing an interactive product for them.
  - Is it to design an efficient system that will allow them to be highly productive in their work?
  - Is it to design a learning tool that will be challenging and motivating?
  - Or, is it something else?
- To identify the objectives, it is suggested to classify them in terms of:
  - **Usability goals** are concerned with meeting specific usability criteria, such as efficiency.
  - **User experience goals** are concerned with explicating the nature of the user experience, for instance, to be aesthetically pleasing.
- It is important to note that the distinction between the two types of goals is not clear-cut since usability is often fundamental to the quality of the user experience and, conversely, aspects of the user experience, such as how it feels and looks, are inextricably linked with how usable the product is.

Usability and user experience goals

- They are distinguished here to help clarify their roles but stress the importance of considering them together when designing for a user experience.

- Historically, HCI was concerned primarily with usability, but it has since become concerned with understanding, designing for, and evaluating a wider range of user experience aspects.

Usability and user experience goals

- Selecting terms to convey a person's feelings, emotions, and so forth can help designers understand the multifaceted nature of the user experience.

- How do usability goals differ from user experience goals?
  - User experience goals are subjective qualities and are concerned with how a system feels to a user.
  - User experience goals differ from the more objective usability goals in that they are concerned with how users experience an interactive product from their perspective, rather than assessing how useful or productive a system is from its own perspective.

Usability and user experience goals

- Are there trade-offs between the two kinds of goals? (for example, can a product be both fun and safe?)
  - Not all usability and user experience goals will be relevant to the design and evaluation of an interactive product being developed.
  - Some combinations will also be incompatible.
    - **Example:** it may not be possible or desirable to design a process control system that is both safe and fun.
  - Recognizing and understanding the nature of the relationship between usability and user experience goals is central to interaction design. It enables designers to become aware of the consequences of pursuing different combinations when designing products and highlighting potential trade-offs and conflicts.
- How easy is it to measure usability versus user experience goals?

Usability goals

- **Usability is broken down into the following six goals:**
  - **1) Effective to use (Effectiveness)**
    - Refers to how good a product is at doing what it is supposed to do.
    - **Question:** Is the product capable of allowing people to learn, carry out their work efficiently, access the information that they need, or buy the goods that they want?
  - **2) Efficient to use (Efficiency)**
    - Refers to the way a product supports users in carrying out their tasks.
    - **Question:** Once users have learned how to use a product to carry out their tasks, can they sustain a high level of productivity?

Usability goals

- **Usability is broken down into the following six goals:**
  - **3) Safe to use (Safety)**
    - Refers to protecting the user from dangerous conditions and undesirable situations.
    - **Question:** What is the range of errors that are possible using the product, and what measures are there to permit users to recover easily from them?
  - **4) Have good utility (Utility)**
    - Refers to the extent to which the product provides the right kind of functionality so that users can do what they need or want to do.
    - **Question:** Does the product provide an appropriate set of functions that will enable users to carry out all of their tasks in the way they want to do them?

Usability goals

- **Usability is broken down into the following six goals:**
  - **5) Easy to learn (Learnability)**
    - Refers to how easy a system is to learn to use. People don't like spending a long time learning how to use a system. This is especially true for interactive products intended for everyday use such as social media, email, or a GPS.
    - **Question:** Is it possible for the user to work out how to use the product by exploring the interface and trying certain actions? How hard will it be to learn the whole set of functions in this way?
  - **6) Easy to remember how to use (Memorability)**
    - Refers to how easy a product is to remember how to use, once learned. This is especially important for interactive products that are used infrequently.
    - **Question:** What types of interface support have been provided to help users remember how to carry out tasks, especially for products and operations they use infrequently?

Usability goals

- In addition to couching usability goals in terms of specific questions, they are turned into **usability criteria**.
  - These are specific objectives that enable the usability of a product to be assessed in terms of how it can improve (or not improve) a user's performance.
- **Examples of commonly used usability criteria are:**
  - Time to complete a task **(Efficiency)**
  - Time to learn a task **(Learnability)**
  - The number of errors made when carrying out a given task over time **(Memorability)**
- These can provide quantitative indicators of the extent to which productivity has increased, or how work, training, or learning have been improved.
  - However, they do not address the overall quality of the user experience.

User experience goals

Desirable aspects

Satisfying	Helpful	Fun
Enjoyable	Motivating	Provocative
Engaging	Challenging	Surprising
Pleasurable	Enhancing sociability	Rewarding
Exciting	Supporting creativity	Emotionally fulfilling
Entertaining	Cognitively stimulating	Experiencing flow

Undesirable aspects

Boring	Unpleasant
Frustrating	Patronizing
Making one feel guilty	Making one feel stupid
Annoying	Cutesy
Childish	Gimmicky

Design principles

- Design principles are used by interaction designers to aid their thinking when designing for the user experience.
- These are generalizable abstractions intended to orient designers toward thinking about different aspects of their designs.
- A well-known example is **feedback**:
  - Products should be designed to provide adequate feedback to the users that informs them about what has already been done so that they know what to do next in the interface.
- Another one that is important is **findability**:
  - This refers to the degree to which a particular object is easy to discover or locate—be it navigating a website, moving through a building, or finding the delete image option on a digital camera.
- Related to this is the principle of **navigability**:
  - Is it obvious what to do and where to go in an interface; are the menus structured in a way that allows the user to move smoothly through them to reach the option they want?

Design principles

- The do’s and don'ts of interaction design.
- What to provide and what not to provide at the interface.
- Derived from a mix of theory-based knowledge, experience, and common-sense.
- The best known design principles are concerned with how to determine what users should see and do when carrying out their tasks using an interactive product.
- **The most common design principles are:**
  - Visibility, Feedback, Constraints, Consistency, and Affordance.

Visibility

- The correct parts must be visible, and they must convey the correct message.
- The more visible functions are, the more likely it is that users will be able to know what to do next.
- Don Norman (1988) describes the controls of a car to emphasize this point. The controls for different operations are clearly visible, such as indicators, headlights, horn, and hazard warning lights, indicating what can be done.
- The relationship between the way the controls have been positioned in the car and what they do makes it easy for the driver to find the appropriate control for the task at hand.

Visibility - poor interface

www.baddesigns.com

- This is a control panel for an elevator
- How does it work?
- Push a button for the floor you want?
- Nothing happens. Push any other button? Still nothing. What do you need to do?
- It is not visible as to what to do!

Visibility - Improving on a poor interface

www.baddesigns.com

…with this elevator, you need to insert your room card in the slot by the buttons to get the elevator to work!

How would you make this action more visible?

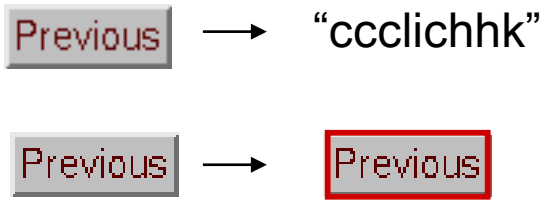
- Make the card reader more obvious
- Provide an auditory message that says what to do (which language?)
- Provide a big label next to the card reader that flashes when someone enters
- Make relevant parts visible
- Make what has to be done obvious

What do I do if I am wearing black?

Invisible automatic controls can make it more difficult to use

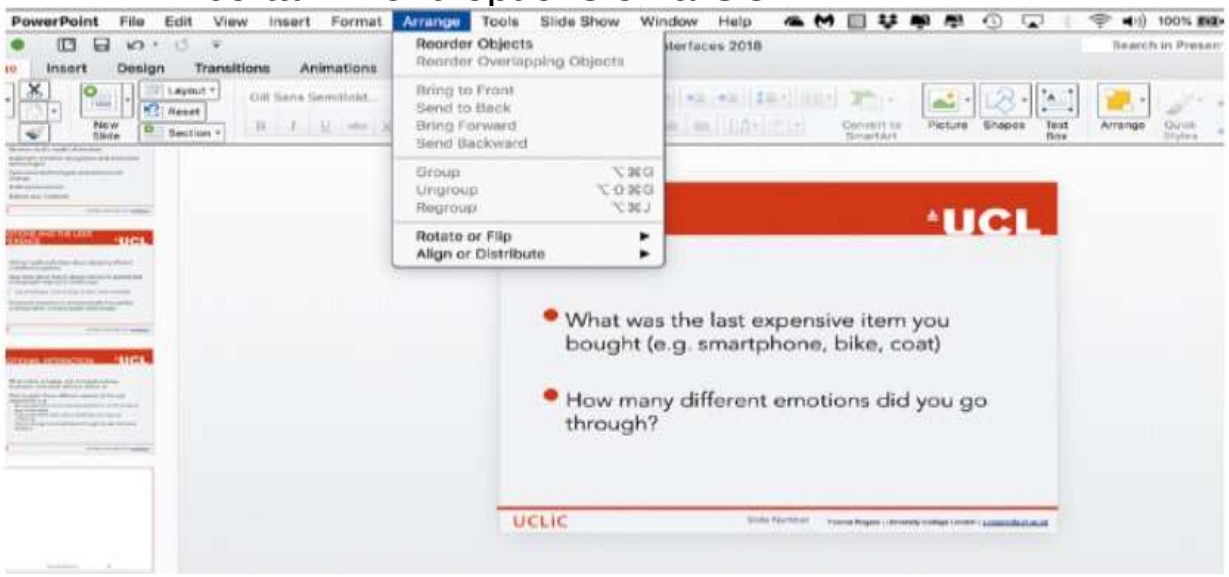
Feedback

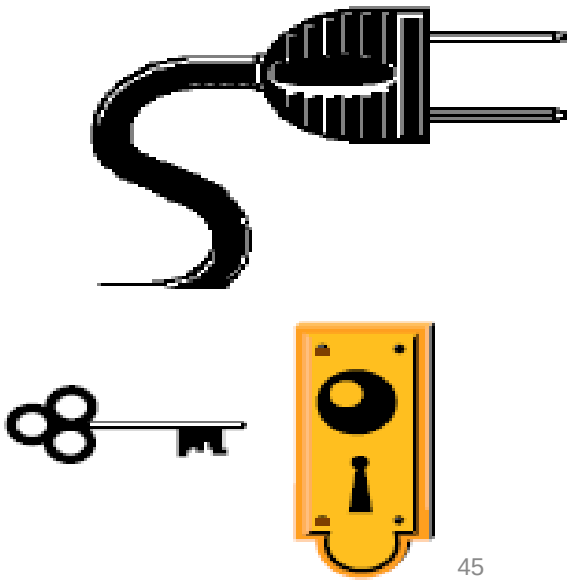
- Feedback involves sending back information about what action has been done and what has been accomplished, allowing the person to continue with the activity.
  - Various kinds of feedback are available for interaction, including:
    - Sound, highlighting, animation, and combinations of these.
  - Deciding which combinations are appropriate for different types of activities and interactivities is central.
  - Using feedback in the right way can also provide the necessary visibility for user interaction.
  - **Example:** when screen button is clicked, it provides sound or red highlight feedback:



Constraints

- Refer to determining ways of restricting the kinds of user interaction that can take place at a given moment.
  - Restricting the possible actions that can be performed.
- Helps prevent user from selecting incorrect options.
- Physical objects can be designed to constrain things.
  - **Example:** there being only one way you can insert a key into a lock or to deactivate certain menu options on a GUI.





Logical or ambiguous design?

www.baddesigns.com

- Where do you plug the mouse?
- Where do you plug the keyboard, in the top or bottom connector?
- Do the color-coded icons help?

How to design them more logically

www.baddesigns.com

(A) provides direct adjacent mapping between icon and connector

(B) provides color coding that associates the connectors with the labels

www.baddesigns.com

Consistency

- Design interfaces to have similar operations and use similar elements for similar tasks. (for example, always use Ctrl key plus first initial of the command for an operation: Ctrl+c, Ctrl+s, Ctrl+o)

- The main benefit is that consistent interfaces are easier to learn and use.

When consistency breaks down

- What happens if there is more than one command starting with the same letter? (for example, save, spelling, select, style)
- You have to find other initials or combinations of keys, thereby breaking the consistency rule (for example, Ctrl+s, Ctrl+Sp, Ctrl+shift+l)
- Increases learning burden on user, making them more prone to errors.

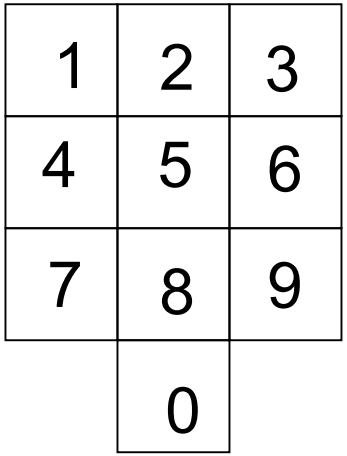
Internal and external consistency

- **Internal consistency** refers to designing operations to behave the same **within an application**
  - Difficult to achieve with complex interfaces
- **External consistency** refers to designing operations, interfaces, and so on to be the same **across applications and devices**
  - Very rarely the case, based on different designer's preference

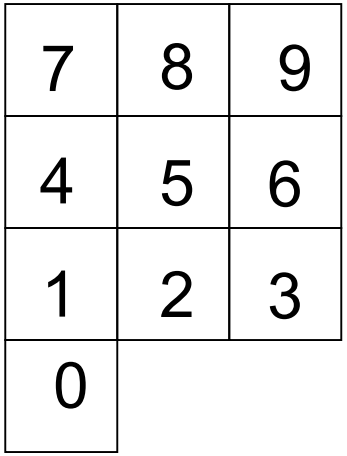
Keypad numbers layout

A case of external inconsistency

(a) phones, remote controls



(b) calculators, computer keypads



Affordances: to give a clue

- Refers to an attribute of an object that allows people to know how to use it. (For example, a mouse button invites pushing, a door handle affords pulling)
- Norman (1988) used the term to discuss the design of everyday objects
- Has since been popularized in interaction design to discuss how to design interface objects (for example, scrollbars to enable moving up and down; icons to click on)

Affordances: to give a clue

- **Appearance indicates how the object should be used:**
  - Chair for sitting
  - Table for placing things on
  - Knobs for turning
  - Slots for inserting things into
  - Buttons for pushing
  - Computers for ???

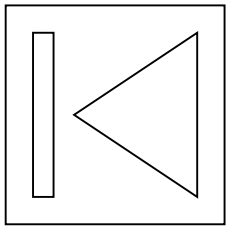
What does “affordance” have to offer interaction design?

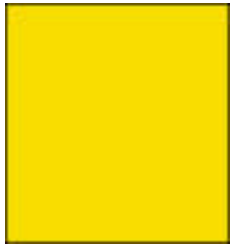
- Interfaces are virtual and do not have affordances like physical objects
- Norman argues that it does not make sense to talk about interfaces in terms of ‘real’ affordances
- Instead, interfaces are better conceptualized as ‘perceived’ affordances:
    - Learned conventions of arbitrary mappings between action and effect at the interface
    - Some mappings are better than others

Activity

Virtual affordances

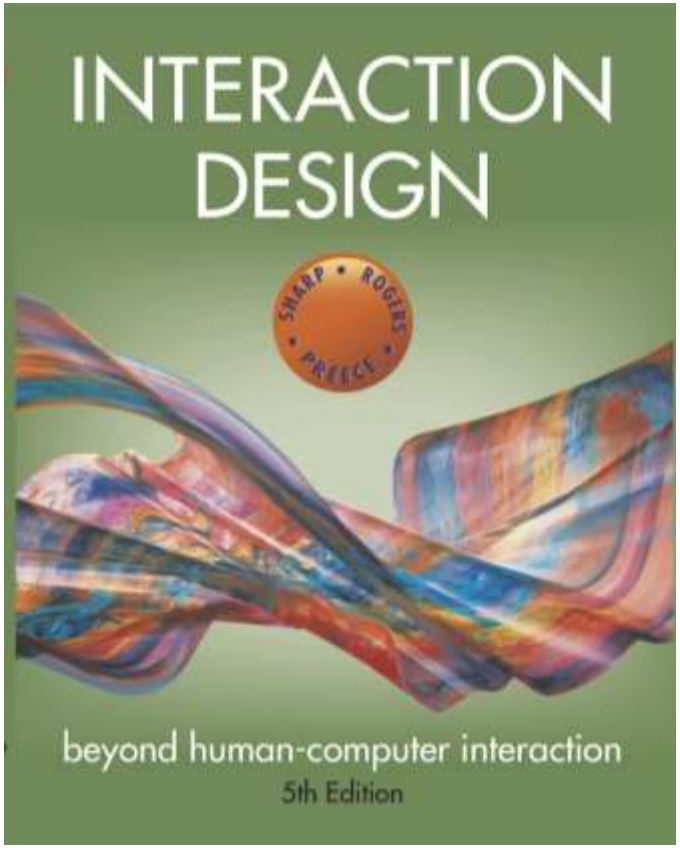
- How do these screen objects afford?
- What if you were a novice user?
- Would you know what to do with them?





Key points

- Interaction design is concerned with designing interactive products to support how people communicate and interact in their everyday and working lives
- It is concerned with how to create quality user experiences for services, devices, and interactive products
- It is multidisciplinary, involving many inputs from wide-reaching disciplines and fields
- Optimizing the interaction between users and interactive products requires consideration of a number of interdependent factors, including context of use, types of activity, UX goals, accessibility, cultural differences, and user groups.
- Design principles, such as feedback and simplicity, are useful heuristics for informing, analyzing, and evaluating aspects of an interactive product.



Chapter 2

THE PROCESS OF INTERACTION DESIGN

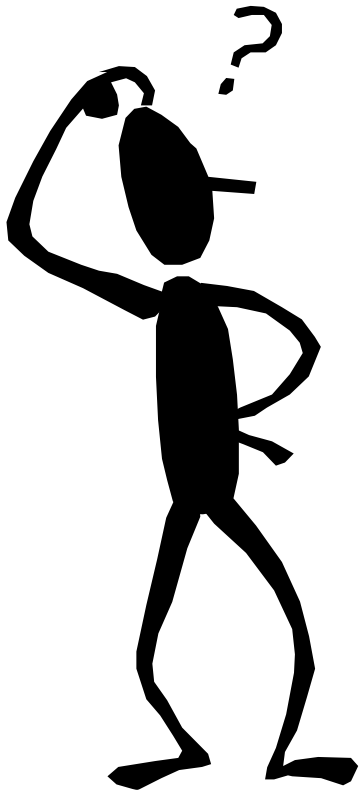
Learning Objectives

- **The main goals of this chapter are to accomplish the following:**
  - Reflect on what interaction design involves.
  - Explain some of the advantages of involving users in development.
  - Explain the main principles of a user-centered approach.
  - Introduce the four basic activities of interaction design and how they are related in a simple lifecycle model.
  - Ask some important questions about the interaction design process and provide the answers.
  - Consider how interaction design activities can be integrated into other development lifecycles.

Overview

What is involved in Interaction Design?

- Understanding the problem space
- Importance of involving users
- Degrees of user involvement
- What is a user-centered approach?
- Four basic activities of interaction design
- A simple lifecycle model for interaction design



Some practical issues

- Who are the users?
- What are the users' needs?
- How to generate alternative designs?
- How to choose among alternative designs?
- How to integrate interaction design activities within other lifecycle models?

Overview

- There are many fields of design, such as graphic design, architectural design, industrial design, and software design.
  - Although each discipline has its own approach to design, there are commonalities.
- The Design Council of the United Kingdom captures these in the double diamond of design, as shown in Figure 2.1.
  - This approach has four phases which are iterated:
    - **Discover:** Designers try to gather insights about the problem.
    - **Define:** Designers develop a clear brief that frames the design challenge.
    - **Develop:** Solutions or concepts are created, prototyped, tested, and iterated.
    - **Deliver:** The resulting project is finalized, produced, and launched.

Overview

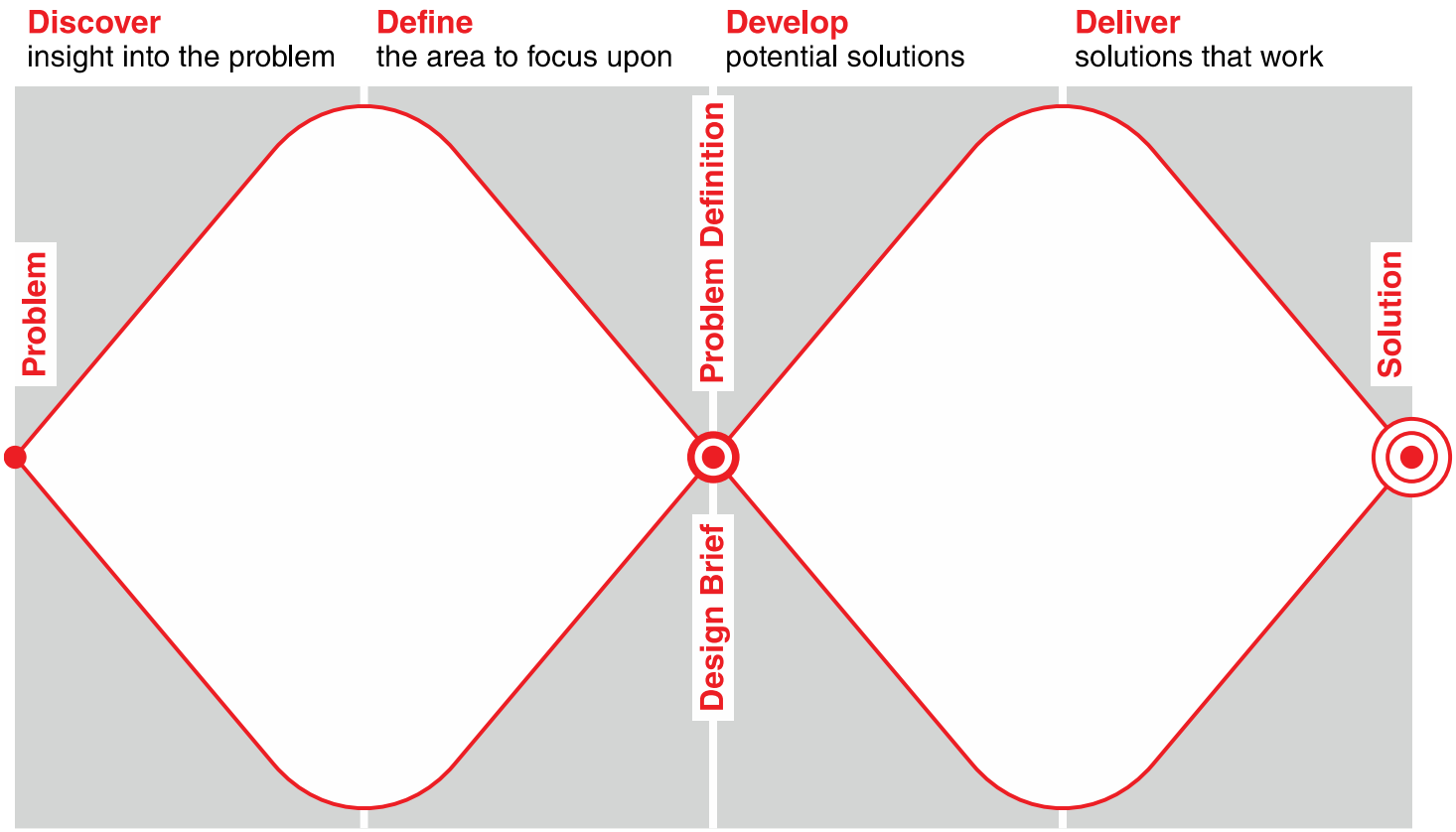


Figure 2.1 The double diamond of design

Source: Adapted from [The Design Process: What is the Double Diamond?]

Overview

- Interaction design also follows these phases, and it is underpinned by the philosophy of user-centered design, that is, involving users throughout development.
- Traditionally, interaction designers begin by doing user research and then sketching their ideas.
  - **But:**
    - Who are the users to be researched?
    - How can they be involved in development?
    - Will they know what they want or need if we just ask them?
    - From where do interaction designers get their ideas, and how do they generate designs?

What is involved in Interaction Design?

- Interaction design has specific activities:
  - Focused on discovering requirements for the product, designing something to fulfill those requirements, and producing prototypes that are then evaluated.
  - Focused attention on users and their goals.
  - Involves trade-offs to balance conflicting requirements
- Generating alternatives and choosing between them is a key principle in most design disciplines and one that is also central to interaction design.
- Dan Saffer (2010) suggests four main approaches to interaction design:
  - User-centered design, Activity-centered design, Systems design, and Genius design.
- Although an individual designer may prefer a particular approach, it is important that the approach for any one design problem is chosen with that design problem in mind.

What is involved in Interaction Design?

- **User-Centered Design**
  - The user knows best and is the guide to the designer; the designer's role is to translate the users' needs and goals into a design solution.
- **Activity-Centered Design**
  - Focuses on the behavior surrounding particular tasks. Users still play a significant role, but it is their behavior rather than their goals and needs that is important.
- **Systems Design**
  - Structured, rigorous, and holistic design approach that focuses on context and is particularly appropriate for complex problems.
  - It is the system (that is, the people, computers, objects, devices, and so on) that the center of attention, while the users' role is to set the goals of the system.
- **Genius Design**
  - Relies largely on the experience and creative flair of a designer.
  - It is also called rapid expert design.
  - Users' role is to validate ideas generated by the designer, and users are not involved during the design process itself.
    - This is not necessarily by choice, but it may be because of limited or no resources for user involvement.

Understanding the problem space

- Deciding what to design is key, and exploring the problem space is one way in which to decide.
  - This is the first phase in the double diamond.
- In the process of creating an interactive product, it can be tempting to begin at the nuts and bolts level of design.
  - This means working out how to design the physical interface and what technologies and interaction styles to use
    - **Example:** Multitouch, Voice, Graphical User Interface, Heads-up Display, Augmented Reality, Gesture-based, …etc.
  - The problem with starting here is that potential users and their context can be misunderstood, and usability and user experience goals can be overlooked.

Understanding the problem space

- While it is certainly necessary at some point to choose which technology to employ and decide how to design the physical aspects, it is better to make these decisions after articulating the nature of the problem space.
- This means understanding:
  - What is currently the user experience or the product?
  - Why is a change needed?
  - How will this change improve user experience?
- The process of articulating the problem space is done as a team effort.
  - Team members will have differing perspectives on it.
    - **Example:** A project manager is concerned about a proposed solution in terms of budgets, timelines, and staffing costs, whereas a software engineer will be thinking about breaking it down into specific technical concepts.
- The implications of pursuing each perspective need to be considered in relation to one another. Although time-consuming and sometimes resulting in disagreements among the design team, the benefits of this process can far outweigh the associated costs:
  - There will be much less chance of incorrect assumptions and unsupported claims

Importance of involving users

- Involving users and others in the design process means that the designs and potential solutions will need to be communicated to people other than the original designer.
- This requires the design to be captured and expressed in a form that allows review, revision, and improvement.
- **There are many ways of doing this:**
  - Produce a series of sketches
  - Write a description in natural language
  - Draw a series of diagrams
  - Build a prototype
    - Limited version of the final product
- Building prototypes is an extremely powerful approach.
  - Because users are unlikely to understand jargon or specialist notations
- In practice, a combination of these techniques is likely to be the most effective.

Importance of involving users

- The best way to ensure that developers gain a good understanding of users' goals, leading to a more appropriate, more usable product, is to involve target users throughout development.
  - However, two aspects unrelated to functionality are equally as important if the product is to be usable and used: 1) Expectation Management and 2) Ownership.
- **1) Expectation management** is the process of making sure that the users' expectations of the new product are realistic. Its purpose is to ensure that there are no surprises for users when the product arrives. If users feel they have been cheated by promises that have not been fulfilled, then this will cause resistance and even rejection.
- Involving users throughout development helps with expectation management because they can see the product's capabilities from an early stage. They will also understand better how it will affect their jobs and lives and why the features are designed that way.
- Adequate and timely training is another technique for managing expectations.
  - If users have the chance to work with the product before it is released through training or hands-on demonstrations of a prerelease version, then they will understand better what to expect when the final product is available.

Importance of involving users

- **2) Ownership**
- Users who are involved and feel that they have contributed to a product's development are more likely to feel a sense of ownership toward it and support its use (Bano et al., 2017).
  - How to involve users, in what roles, and for how long, needs careful planning.
- Make the users active stakeholders
- More likely to forgive or accept problems
- Can make a big difference in acceptance and success of product

Degrees of user involvement

- Different degrees of user involvement are possible, ranging from fully engaged throughout all iterations of the development process to targeted participation in specific activities and from small groups of individual users in face-to-face contexts to hundreds of thousands of potential users and stakeholders online.
- Member of the design team:
  - Full time: constant input, but lose touch with users
  - Part time: patchy input, and very stressful
  - Short term: inconsistent across project life
  - Long term: consistent, but lose touch with users
- Face-to-face group or individual activities
- Online contributions from thousands of users
  - Online Feedback Exchange (OFE) systems
    - Increasingly used to test design concepts with millions of target users before going to market
  - Crowdsourcing design ideas
    - Range of different people are encouraged to contribute, and this can include any and all of the stakeholders.
  - Citizen science
    - The goal of which is to engage a population—civic or otherwise—with the aim of promoting empowerment through technology.
- User involvement after product release

What is a user-centered approach?

- User-centered approach is based on three principles:
  - **Early focus on users and tasks:** directly studying cognitive, behavioral, anthropomorphic, and attitudinal characteristics
  - **Empirical measurement:** users' reactions and performance to scenarios, manuals, simulations, and prototypes are observed, recorded, and analysed.
    - Where possible, specific usability and user experience goals should be identified, clearly documented, and agreed upon at the beginning of the project.
  - **Iterative design:** when problems are found in user testing, fix them and carry out more tests

What is a user-centered approach?

- **Early focus on users and tasks:**
- This principle can be expanded and clarified through the following five further principles:
  - **1.** Users' tasks and goals are the driving force behind the development.
  - **2.** Users' behavior and context of use are studied, and the system is designed to support them.
  - **3.** Users' characteristics are captured and designed for.
  - **4.** Users are consulted throughout development from earliest phases to the latest.
  - **5.** All design decisions are taken within the context of the users, their activities, and their
  - environment.

Four basic activities of Interaction Design

1. **Discovering requirements**

- This activity covers the left side of the double diamond of design, and it is focused on discovering something new about the world and defining what will be developed. In the case of interaction design, this includes understanding the target users and the support an interactive product could usefully provide.

2. **Designing alternatives**

- This is the core activity of designing and is part of the Develop phase of the double diamond: proposing ideas for meeting the requirements. For interaction design, this activity can be viewed as two subactivities: conceptual design and concrete design.
  - **Conceptual design** involves producing the conceptual model for the product, and a conceptual model describes an abstraction outlining what people can do with a product and what concepts are needed to understand how to interact with it.
  - **Concrete design** considers the detail of the product including the colors, sounds, and images to use, menu design, and icon design.

Four basic activities of Interaction Design

3. Prototyping alternative designs

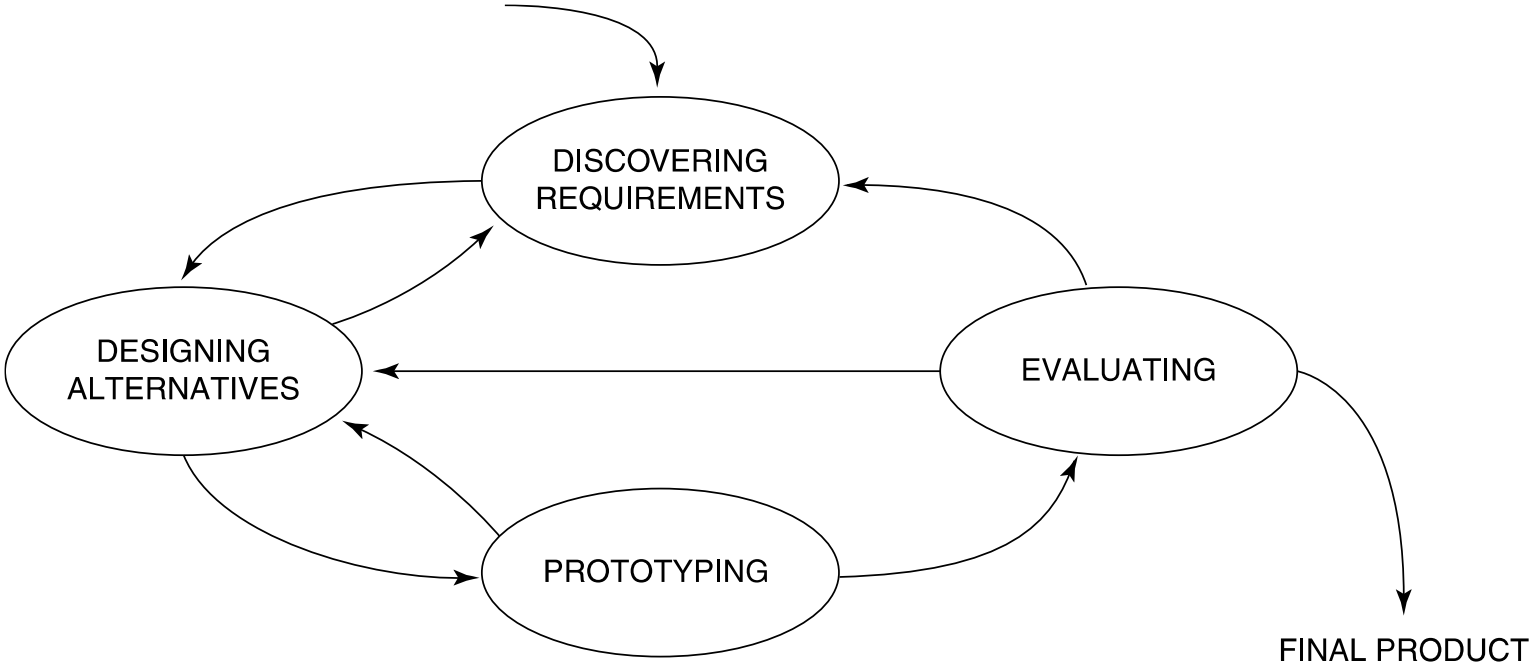
- Prototyping is also part of the Develop phase of the double diamond. Interaction design involves designing the behavior of interactive products as well as their look and feel. The most effective way for users to evaluate such designs is to interact with them, and this can be achieved through prototyping.

4. Evaluating product and its user experience throughout the process

- Evaluating is also part of the Develop phase of the double diamond. It is the process of determining the usability and acceptability of the product or design measured in terms of a variety of usability and user-experience criteria. Evaluation does not replace activities concerned with quality assurance and testing to make sure that the final product is fit for its intended purpose, but it complements and enhances them.

A simple interaction design lifecycle model

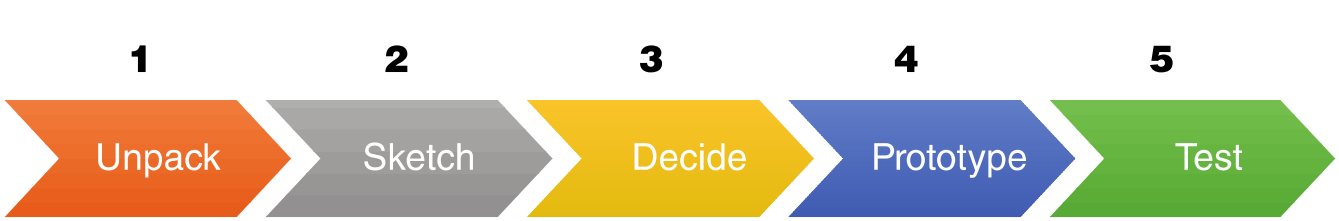
Exemplifies a user-centered design approach

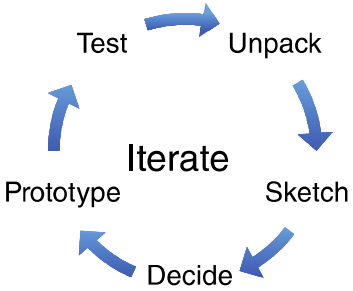


Another lifecycle model: Google Design Sprints (Knapp et al., 2016)

- Google Ventures has developed a structured approach to design that supports rapid ideation and testing of potential solutions to a design challenge. This is called the *Google Design Sprint*.
- A sprint is divided into five phases, and each phase is completed in a day. This means that in five days, you can go from a design challenge to a solution that has been tested with customers.

Google Design Sprint





Source: Google Design Sprints (used courtesy of Agile Marketing)

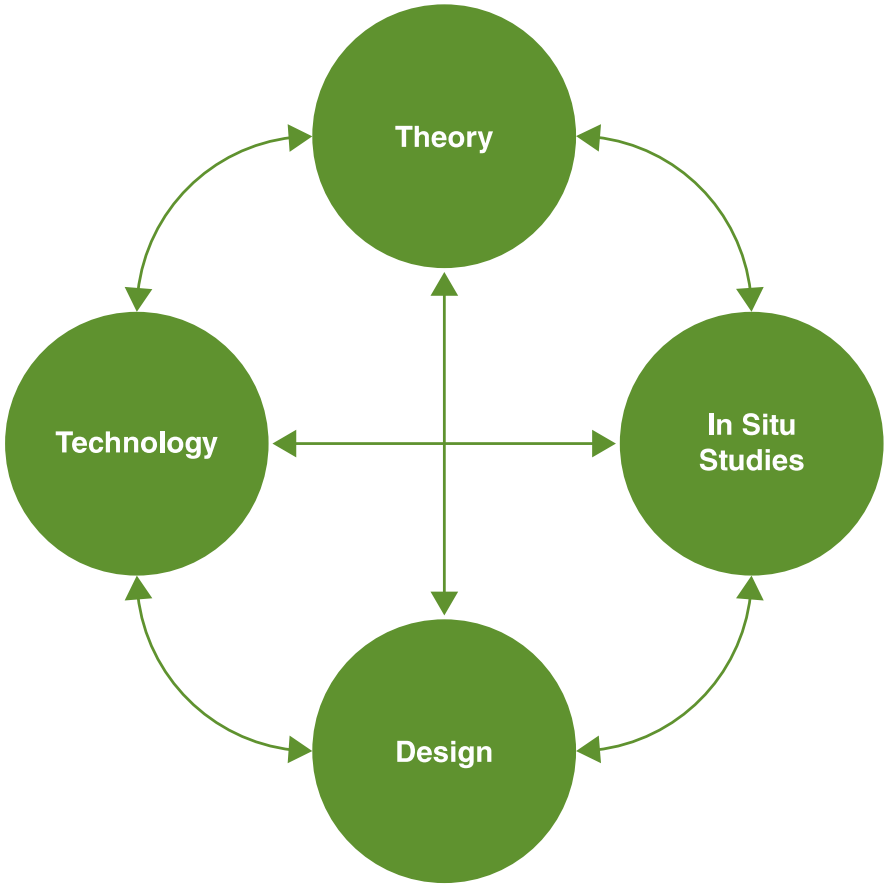
Another lifecycle model: Google Design Sprints (Knapp et al., 2016)

- **Unpack**
  - Day 1 focuses on making a map of the challenge and choosing a target, that is, a part of the challenge that can be achieved in a week.
- **Sketch Competing Solutions**
  - Day 2 focuses on generating solutions, with an emphasis on sketching and individual creativity rather than group brainstorming.
- **Decide on the Best**
  - Day 3 focuses on critiquing the solutions generated on Day 1, choosing the one most likely to meet the sprint's challenge, and producing a storyboard. Whichever solution is chosen, the decider needs to support the design.
- **Build a Realistic Prototype**
  - Day 4 focuses on turning the storyboard into a realistic prototype, that is, something on which customers can provide feedback. We discuss prototyping further in Chapter 12.
- **Test with Target Customers**
  - Day 5 focuses on getting feedback from five customers and learning from their reactions.

Another lifecycle model: Research in the Wild (Rogers and Marshall, 2017)

- Research in the wild (RITW) develops technology solutions in everyday living by creating and evaluating new technologies and experiences *in situ*.
- The approach supports designing prototypes in which researchers often experiment with new technological possibilities that can change and even disrupt behavior, rather than ones that fit in with existing practices.
- The results of RITW studies can be used to challenge assumptions about technology and human behavior in the real world and to inform the re-thinking of HCI theories.
- The perspective taken by RITW studies is to observe how people react to technology and how they change and integrate it into their everyday lives.

Another lifecycle model: Research in the Wild (Rogers and Marshall, 2017)



A framework for research in the wild studies

Source: Rogers and Marshall, 2017, p6. (used courtesy of Morgan and Claypool)

Another lifecycle model: Research in the Wild (Rogers and Marshall, 2017)

- In terms of the four activities introduced earlier, this framework focuses on designing, prototyping, and evaluating technology and ideas and is one way in which requirements may be discovered. It also considers relevant theory since often the purpose of an RITW study is to investigate a theory, idea, concept, or observation (Ethnography + Focused Ethnography). Any one RITW study may emphasize the elements of the framework to a different degree.
- **Technology:**
  - Concerned with appropriating existing infrastructures/devices (e.g., Internet of Things toolkit, mobile app) *in situ* or developing new ones for a given setting (e.g., a novel public display).
- **Design:**
  - Covers the design space of an experience (e.g., iteratively creating a collaborative travel planning tool for families to use or an augmented reality game for playing outdoors).
- **In situ study:**
  - Concerned with evaluating *in situ* an existing device/tool/service or novel research-based prototype when placed in various settings or given to someone to use over a period of time.
- **Theory:**
  - Investigating a theory, idea, concept or observation about a behavior, setting or other phenomenon using existing ones or developing a new one or extending an existing one.

Some practical issues

- Who are the users?
- What are the users' needs?
- How to generate alternative designs?
- How to choose among alternatives?
- How to integrate interaction design activities with other lifecycle models?

Who are the users/stakeholders?

- Identifying users may seem like a straightforward activity, but it can be harder than you think.
- 382 distinct types of users for smartphone apps (Sha Zhao et al, 2016)
  - Many products are intended for use by large sections of the population, so user is “everybody”
  - More targeted products are associated with specific roles
- Not as obvious as you think:
  - — Those who interact directly with the product
  - — Those who manage direct users
  - — Those who receive output from the product
  - — Those who test the system
  - — Those who make the purchasing decision
  - — Those who use competitor’s products ???

Who are the users/stakeholders?

- Three categories of user:
  - **Primary:** frequent hands-on users of the system
  - **Secondary:** occasional or via someone else;
  - **Tertiary:** affected by the introduction of the system, or will influence its purchase.
- We use a wider term called “Stakeholders”
  - Larger than the group of direct users
  - Identifying stakeholders helps identify groups to include in interaction design activities
  - There is a surprisingly wide collection of people who all have a stake in the development of a successful product. These people are called stakeholders.
  - Stakeholders are "people or organizations who will be affected by the system and who have a direct or indirect influence on the system requirements" (Kotonya and Sommerville, 1998).

Who are the users/stakeholders?

- What are their capabilities?
  - Humans vary in many dimensions!
- Some examples are:
  - **Size of hands** may affect the size and positioning of input buttons;
  - **Motor abilities** may affect the suitability of certain input and output devices;
  - **Height** if designing a physical kiosk;
  - **Strength** - a child's toy requires little strength to operate, but greater strength to change batteries

What are the users' needs?

- Users rarely know what is possible
- Users can't tell you what they 'need' to help them achieve their goals
- Instead, look at existing tasks:
  - their context
  - what information do they require?
  - who collaborates to achieve the task?
  - why is the task achieved the way it is?
- Envisioned tasks:
  - can be rooted in existing behaviour
  - can be described as future scenarios
- Instead:
  - Explore the problem space
  - Investigate who are the users
  - Investigate user activities to see what can be improved
  - Try out ideas with potential users
- Focus on peoples' goals, usability, and user experience goals, rather than expect stakeholders to articulate requirements

What are the users' needs?

- If a product is a **new invention**, then it can be difficult to identify the users and representative tasks for them;
- E.g.: before microwave ovens were invented, there were no users to consult about requirements and there were no representative tasks to identify.
- Those developing the oven had to imagine who might want to use such an oven and what they might want to do with it.

How to generate alternatives

- Humans tend to stick with something that works
- Considering alternatives helps identify better designs
- Where do alternative designs come from?
  - ‘Flair and creativity’: research and synthesis
  - Cross-fertilization of ideas from different perspectives
  - Users can generate different designs
  - Product evolution based on changing use
  - Seek inspiration: similar products and domain, or different products and domain
- Balancing constraints and trade-offs

How to choose among alternatives

- Interaction design focuses on externally-visible and measurable behavior
- Technical feasibility
- Evaluation with users or peers
  - Prototypes not static documentation because behavior is key
- A/B Testing
  - Online method to inform choice between alternatives
  - Nontrivial to set appropriate metrics and choose user group sets
- Quality thresholds
  - Different stakeholder groups have different quality thresholds
  - Usability and user experience goals lead to relevant criteria

How to integrate interaction design activities within other models

- Integrating interaction design activities in lifecycle models from other disciplines requires careful planning
- Software development lifecycle models are prominent
- Integrating with agile software development is promising because:
  - It incorporates tight iterations
  - It champions early and regular feedback
  - It handles emergent requirements
  - It aims to strike a balance between flexibility and structure

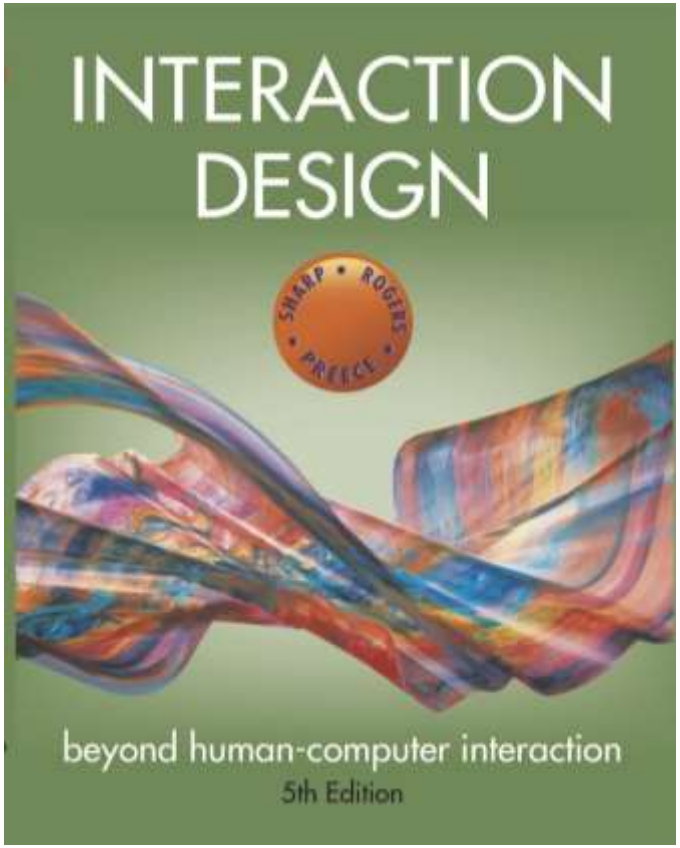
Some key points

Four basic activities in interaction design process

- Discovering requirements
- Designing alternatives
- Prototyping
- Evaluating

User-centered design rests on three principles

- Early focus on users and tasks
- Empirical measurement using quantifiable and measurable usability criteria
- Iterative design



Chapter 3

CONCEPTUALIZING INTERACTION DESIGN?

Learning Objectives

- **The main goals of this chapter are to accomplish the following:**
  - Explain how to conceptualize interaction.
  - Describe what a conceptual model is and how to begin to formulate one.
  - Discuss the use of interface metaphors as part of a conceptual model.
  - Outline the core interaction types for informing the development of a conceptual model.
  - Introduce paradigms, visions, theories, models, and frameworks informing interaction design.

Conceptualizing design

- When coming up with new ideas as part of a design project, it is important to conceptualize them in terms of what the proposed product will do.
  - This is referred to as creating a **Proof of Concept**.
- Why the need to conceptualizing design?
  - Reality check to scrutinize vague ideas and assumptions about the benefits of the proposed product in terms of their feasibility
  - How realistic is it to develop what they have suggested?
  - How desirable and useful will it actually be?
  - Enable designers to begin articulating what the basic building blocks will be when developing the product.
  - From a user experience (UX) perspective, it can lead to better clarity, forcing designers to explain how users will understand, learn about, and interact with the product.

Assumptions and claims

- Write down your assumptions and claims when coming up with a new design
- Try to defend and support them by what they will provide
- Those that are difficult to articulate
  - Can highlight what ideas are vague or unrealistic
  - Identify human activities and interactivities that are problematic
- Iteratively work out how the design ideas might be improved

What is an assumption?

- **Assumption** is taking something for granted when it needs further investigation
  - **Example:** people now want an entertainment and navigation system in their cars.

Technotopic Narratives and Networked Subjects: Preparations for Everyday Life in Cooltown

What is a claim?

- **Claim** is stating something to be true when it is still open to question.
  - **Example:** a multimodal style of interaction for controlling this system “GPS” — one that involves speaking or gesturing while driving — is perfectly safe.

Activity: How will enabling robot waiters to speak to customers enhance their experience?

- Consider the bright idea that a designer has of creating a voice-assisted mobile robot that can help waiters in a restaurant take orders and deliver meals to customers (see Figure 3.1).

Source: Xinhua, Guo Cheng

Figure 3.1 A nonspeaking robot waiter in Shanghai. What would be gained if it could also talk with customers?

Activity: How will enabling robot waiters to speak to customers enhance their experience?

- The first question to ask is:
  - Why?
  - What problem would this address?
- The designer might say a number of benefits such as:
  - The robot could help take orders and entertain customers by having a conversation with them at the table.
  - The robot could make recommendations that can be customized to different customers, such as restless children or fussy eaters.
- However, none of these addresses an actual problem. Rather, they are couched in terms of the putative benefits of the new solution.
  - These are just assumptions.
- In contrast, an actual problem identified might be the following:
  - It is difficult to recruit good wait staff who provide the level of customer service to which we have become accustomed.

Activity: How will enabling robot waiters to speak to customers enhance their experience?

- Having worked through a problem space, it is important to generate a set of research questions that need to be addressed, when considering how to design a robot voice interface to wait on customers.
- These might include the following:
  - How intelligent does it have to be?
  - How would it need to move to appear to be talking?
  - What would the customers think of it?
  - Would they think it is too gimmicky and get easily tired of it? Or, would it always be a pleasure for them to engage with the robot, not knowing what it would say on each new visit to the restaurant?
  - Could it be designed to be a grumpy extrovert or a funny waiter?
  - What might be the limitations of this voice-assisted approach?

Working through assumptions

- Many unknowns need to be considered in the initial stages of a design project, especially if it is a new product that is being proposed.
- As part of this process, it can be useful to show:
  - Where do your novel ideas come from?
  - What sources of inspiration were used?
  - Is there any theory or research that can be used to inform and support the ideas?
- During the early ideation process
  - Ask questions, reconsider assumptions, and articulate concerns

A framework for analyzing the problem space

- Explaining people's assumptions and claims about why they think something might be a good idea (or not) enables the design team as a whole to view multiple perspectives on the problem space and, in so doing, reveals conflicting and problematic ones.
- The following framework is intended to provide a set of core questions to aid design teams in this process:
  - Are there problems with an existing product or user experience? If so, what are they?
  - Why do you think there are problems?
  - What evidence do you have to support the existence of these problems?
  - How do you think your proposed design ideas might overcome these problems?

Activity

- What were the assumptions and claims made about watching 3D TV?

Figure 3.2 A family watching 3D TV

Source: Andrey Popov, Shutterstock

Assumptions and claims: how realistic?

- There was no existing problem to overcome
  - What was being proposed was a new way of experiencing TV
- An assumption:
  - People would not mind wearing the glasses that were needed to see in 3D, nor would they mind paying a lot more for a new 3D-enabled TV screen, because of the new experience.
- A claim:
  - People would really enjoy the enhanced clarity and color detail provided by 3D, based on the favorable feedback received worldwide when viewing 3D films, such as *Avatar*, at a cinema.

Benefits of conceptualizing the design space

- Making clear what one's assumptions are about a problem and the claims being made about potential solutions should be carried out early on and throughout a project.
- Design teams also need to work out how best to conceptualize the design space.
- Primarily, this involves articulating the proposed solution as a conceptual model with respect to the user experience.
- The benefits of conceptualizing the design space in this way are as follows:
- **1) Orientation**
  - Enabling the design team to ask specific kinds of questions about how the conceptual model will be understood by the targeted users.
- **2) Open-minded**
  - Allowing the team to explore a range of different ideas to address the problems identified.
- **3) Common ground**
  - Allowing the design team to establish a set of common terms that all can understand and agree upon, reducing the chance of misunderstandings and confusion arising later.

From problem space to design space

- Having a good understanding of the problem space can help inform the design space.
  - **Example:** what kind of interface, behavior, functionality to provide
- Before deciding upon these, it is important to develop a conceptual model.

Conceptual model

- A **model** is a simplified description of a system or process that helps describe how it works.
- A particular kind of model used in interaction design intended to articulate the problem and design space—the **conceptual model**.
- A conceptual model is:
  - "…a high-level description of how a system is organized and operates" (Johnson and Henderson, 2002, p26)
- A conceptual model enables:
  - "…designers to straighten out their thinking before they start laying out their widgets" (Johnson and Henderson, 2002, p28)
- Provides a working strategy and framework of general concepts and their interrelations

Conceptual model

- A Classic Conceptual Model: The Xerox Star
  - The Star interface, developed by Xerox in 1981 (see Figure 3.4), revolutionized the way that interfaces were designed for personal computing (Smith et al., 1982; Miller and Johnson, 1996).

Figure 3.4 The Xerox Star
Source: Used courtesy of Xerox

- Watch this video:
  - https://www.youtube.com/watch?v=Cn4vC80Pv6Q

Components of the conceptual model

- 1) Metaphors and analogies
  - Convey to people how to understand what a product is used for and how to use it for an activity (for example browsing and bookmarking).
- 2) Concepts to which people are exposed through the product,
  - Including the task-domain objects they create and manipulate, their attributes, and the operations that can be performed on them (such as saving, revisiting, and organizing).
- 3) The relationships between those concepts
  - For example: whether one object contains another.
- 4) The mappings between the concepts and the user experience the product is designed to support or invoke
  - For example: one can revisit a page through looking at a list of visited sites, most-frequently visited, or saved websites.

Components of the conceptual model

- How the various metaphors, concepts, and their relationships are organized determines the user experience.
- By explaining these, the design team can debate the merits of providing different methods and how they support the main concepts, for example, saving, revisiting, categorizing, reorganizing, and their mapping to the task domain.
- They can also begin discussing whether a new overall metaphor may be preferable that combines the activities of browsing, searching, and revisiting.
- In turn, this can lead the design team to articulate the kinds of relationships between them, such as containership.
  - **For Example:** what is the best way to sort and revisit saved pages, and how many and what types of containers should be used (for example, folders, bars, or panes)? The same enumeration of concepts can be repeated for other functions of the web browser—both current and new.
- In so doing, the design team can begin to work out systematically what will be the simplest and most effective and memorable way of supporting users while browsing the Internet.

First steps in formulating a conceptual model

- What will the users be doing when carrying out their tasks?
- How will the system support these?
- What kind of interface metaphor, if any, will be appropriate?
- What kinds of interaction modes and styles to use?
  - Always keep in mind when making design decisions how the user will understand the underlying conceptual model

Conceptual models

- Many kinds and ways of classifying them
- The best conceptual models are often those that appear:
  - Obvious and simple
  - The operations they support are intuitive to use

Interface metaphors

- Metaphors are considered to be a central component of a conceptual model.
- Metaphors provide a structure that is similar in some way to aspects of a familiar entity (or entities), but they also have their own behaviors and properties.
- More specifically, an **interface metaphor** is one that is instantiated in some way as part of the user interface, such as the desktop metaphor.
- Another well-known one is the **search engine**, originally coined in the early 1990s to refer to a software tool that indexed and retrieved files remotely from the Internet using various algorithms to match terms selected by the user.

Interface metaphors

- Interface designed to be similar to a physical entity but also has own properties
  - For example: desktop metaphor, and web portals
- Can be based on activity, object, or a combination of both
- Exploit user's familiar knowledge, helping them to understand 'the unfamiliar'
- Conjures up the essence of the unfamiliar activity, enabling users to leverage this to understand more aspects of the unfamiliar functionality

Examples of interface metaphors

- Conceptualizing what users are doing
  - For instance, surfing the Web
- A conceptual model instantiated at the interface
  - For example, the desktop metaphor
- Visualizing an operation
  - For instance, an icon of a shopping cart into which the user places items

Examples of interface metaphors

- Interface metaphors are intended to provide familiar entities that enable people readily to understand the underlying conceptual model and know what to do at the interface. However, they can also contravene people's expectations about how things should be, such as the recycle bin (trash can) that sits on the desktop.
- Logically and culturally (meaning, in the real world), it should be placed under the desk. But users would not have been able to see it because it would have been hidden by the desktop surface. So, it needed to go on the desktop. While some users found this irksome, most did not find it to be a problem. Once they understood why the recycle bin icon was on the desktop, they simply accepted it being there.
- An interface metaphor that has become popular in the last few years is the **card**.
- Many of the social media apps, such as Facebook, Twitter, and Pinterest, present their content on cards.
- Cards have a familiar form, having been around for a long time.
- Just think of how many kinds there are: playing cards, business cards, birthday cards, credit cards, and postcards to name a few.

Examples of interface metaphors

- Cards have strong associations, providing an intuitive way of organizing limited content that is “card sized.” They can easily be flicked through, sorted, and themed.
- Cards structure content into meaningful chunks, similar to how paragraphs are used to chunk a set of related sentences into distinct sections (Babich, 2016).
- In the context of the smartphone interface, the Google Now card provides short snippets of useful information.
- This appears on and moves across the screen in the way people would expect a real card to do—in a lightweight, paper-based sort of way.
- The elements are also structured to appear as if they were on a card of a fixed size, rather than, say, in a scrolling web page.

The card metaphor

- The card is a very popular UI. Why?
  - It has familiar form factor
  - It can easily be flicked through, sorted, and themed
  - It structures content into meaningful chunks (similar to how paragraphs are used to chunk a set of related sentences into distinct sections)
  - Its material properties give the appearance of the surface of paper

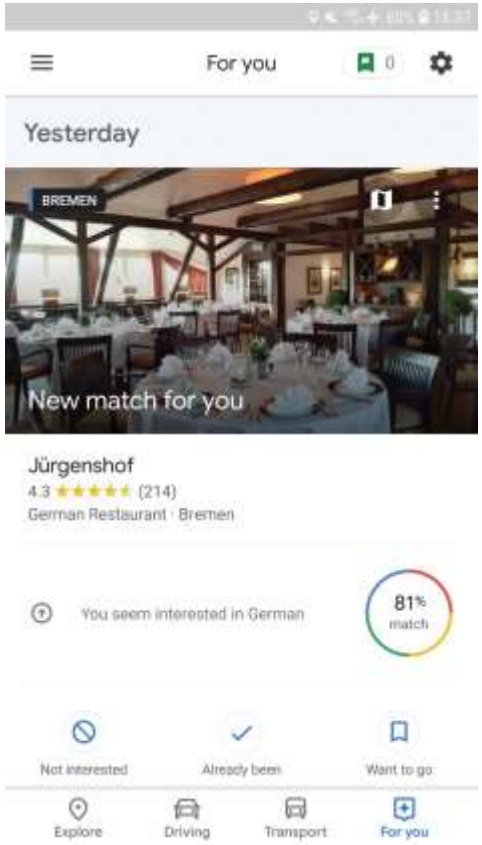


Figure 3.5 Google Now card for restaurant recommendation in Germany

Source: Johannes Shonning

Benefits of interface metaphors

- Makes learning new systems easier
- Helps users understand the underlying conceptual model
- Can be very innovative and enable the realm of computers and their applications to be made more accessible to a greater diversity of users

Problems with interface metaphors

- Break conventional and cultural rules
  - For instance, recycle bin placed on desktop
- Can constrain designers in the way that they conceptualize a problem space
- Conflicts with design principles
- Forces users to understand only the system in terms of the metaphor
- Designers can inadvertently use bad existing designs and transfer the bad parts over
- Limits designers' imagination in coming up with new conceptual models

Activity

- Describe the components of the conceptual model underlying most online shopping websites, for example:
  - Shopping cart
  - Proceeding to check-out
  - 1-click
  - Gift wrapping
  - Cash register

Interaction types

- **1) Instructing**
  - Where users issue instructions to a system.
  - This can be done in a number of ways, including typing in commands, selecting options from menus in a windows environment or on a multitouch screen, speaking aloud commands, gesturing, pressing buttons, or using a combination of function keys.
- **2) Conversing**
  - Where users have a dialog with a system. Users can speak via an interface or type in questions to which the system replies via text or speech output.
- **3) Manipulating**
  - Where users interact with objects in a virtual or physical space by manipulating them (for instance, opening, holding, closing, and placing). Users can hone their familiar knowledge of how to interact with objects.

Interaction types

- **4) Exploring**
  - Where users move through a virtual environment or a physical space.
    - Virtual environments include 3D worlds and augmented and virtual reality systems. They enable users to hone their familiar knowledge by physically moving around.
    - Physical spaces that use sensor-based technologies include smart rooms and ambient environments, also enabling people to capitalize on familiarity.
- **5) Responding**
  - Where the system initiates the interaction and the user chooses whether to respond.
    - **Example:** Proactive mobile location-based technology can alert people to points of interest. They can choose to look at the information popping up on their phone or ignore it. An example is the Google Now Card, shown in Figure 3.5, which pops up a restaurant recommendation for the user to contemplate when they are walking nearby.

1. Instructing

- Where users instruct a system and tell it what to do
  - For example: Tell the time, print a file, or save a file
- Very common conceptual model underlying a diversity of devices and systems
  - For instance: Word processors, VCRs, and vending machines
- The main benefit is that instructing supports quick and efficient interaction
  - Good for repetitive kinds of actions performed on multiple objects

Which is easiest and why?

2. Conversing

- Underlying model of having a conversation with another human
- Ranges from simple voice recognition menu-driven systems to more complex ‘natural language’ dialogs
- Examples include timetables, search engines, advice-giving systems, and help systems
- Also virtual agents, chatbots, toys, and pet robots designed to converse with you

Pros and cons of conversational model

- Allows users, especially novices, to interact with a system in a way that is familiar to them
  - Can make them feel comfortable, at ease, and less scared
- Misunderstandings can arise when the system does not know how to parse what the user says
  - For example, voice assistants can misunderstand what children say

**“If you’d like to press 1, press 3.
If you’d like to press 3, press 8.
If you’d like to press 8, press 5...”**

3. Manipulating

- Involves dragging, selecting, opening, closing and zooming actions on virtual objects
- Exploit's users' knowledge of how they move and manipulate in the physical world
- Can involve actions using physical controllers (for example, Nintendo Wii) or air gestures (such as, Microsoft Kinect) to control the movements of an on-screen avatar
- Tagged physical objects (for instance, balls) that are manipulated in a physical world result in physical/digital events (such as animation)

Direct Manipulation (DM)

- Ben Shneiderman (1983) coined the term DM
- Three core properties:
  - Continuous representation of objects and actions of interest
  - Physical actions and button pressing instead of issuing commands with complex syntax
  - Rapid reversible actions with immediate feedback on object of interest

Benefits of direct manipulation

- Novices can learn the basic functionality quickly
- Experienced users can work extremely rapidly to carry out a wide range of tasks−even defining new functions
- Intermittent users can retain operational concepts over time
- Error messages rarely needed
- Users can immediately see if their actions are furthering their goals, and if not, do something else
- Users experience less anxiety
- Users gain confidence and mastery and feel in control

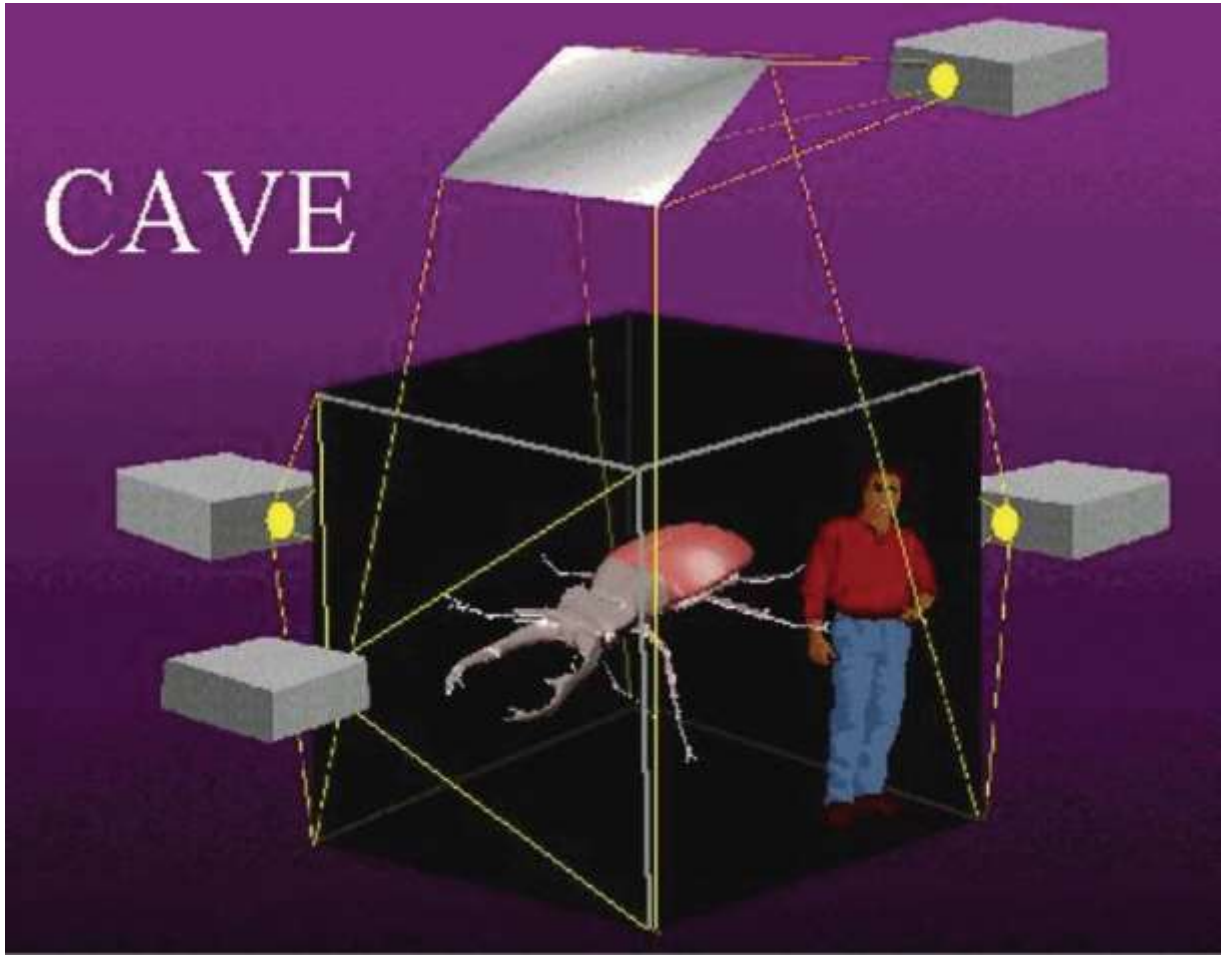
Disadvantages of DM

- Some people take the metaphor of direct manipulation too literally
- Not all tasks can be described by objects, and not all actions can be done directly
- Some tasks are better achieved through delegating, for example, spell checking
- Can become screen space ‘gobblers’
- Moving a cursor using a mouse or touchpad can be slower than pressing function keys to do the same actions

4. Exploring

- Involves moving through virtual or physical environments
  - Users can explore aspects of a virtual 3D environment
  - Physical environments can also be embedded with sensors that when detect the presence of someone will trigger digital or physical events to happen
- Many examples of virtual environments, including cities, parks, buildings, rooms, and datasets
  - Enable users to fly over them and zoom in and out of different parts

Seeing things larger than life in VR



Cyber-Insects in the CAVE *Source*: Alexei A. Sharov

Exploring data in VR

Image courtesy of Kalev Leetaru, National Center for Supercomputing Applications, University of Illinois.

5. Responding

- System takes the initiative to alert user to something that it “thinks” is of interest
- System does this by:
  - Detecting the location and-or presence of someone in a vicinity and notifies them on their phone or watch,
  - What it has learned from their repeated behaviors
- Examples:
  - Alerts the user of a nearby coffee bar where some friends are meeting
  - User’s fitness tracker notifies them of a milestone reached
- Automatic system response without any requests made by the user

This type suggested by Christopher Lueg et al. (2018)

Potential cons of system-initiated notifications

- Can get tiresome or frustrating if too many notifications or the system gets it wrong
- What does it do when it gets something wrong?
  - Does it apologize?
  - Does it allow the user to correct the advise or information?

Choosing an interaction type

- Direct manipulation is good for ‘doing’ types of tasks, for example, designing, drawing, flying, driving, or sizing windows
- Issuing instructions is good for repetitive tasks, for example, spell-checking and file management
- Having a conversation is good for certain services, for instance, finding information or requesting music
- Hybrid conceptual models are good for supporting multiple ways of carrying out the same actions

Difference between interaction types and interface styles

Interaction type:

- A description of what the user is doing when interacting with a system, for example, instructing, talking, browsing, or responding

Interface style:

- The kind of interface used to support the interaction, for instance, command, menu-based, gesture, or voice

Many kinds of interface styles available (see Chapter 7)…

- Command
- Speech
- Data-entry
- Form fill-in
- Query
- Graphical
- Web
- Pen
- Augmented reality
- Gesture

Other sources

Conceptual knowledge that is used to inform design and guide research include:

- Paradigms
- Visions
- Theories
- Models
- Frameworks

Paradigm

- Inspiration for a conceptual model
- General approach adopted by a community for carrying out research
  - Shared assumptions, concepts, values, and practices
  - For example, desktop, ubiquitous computing, in the wild

Examples of new paradigms in HCI

- Ubiquitous computing
- Pervasive computing
- Wearable computing
- Internet of Things (IoT)

Visions

- A driving force that frames research and development
- Invites people to imagine what life will be like in 10, 15, or 20 years' time
  - For example, Apple's 1987 knowledge navigator
  - Smart cities, smart health
  - Human-centered AI
- Provide concrete scenarios of how society can use the next generation of imagined technologies
- Also raise ethical questions such as, privacy and trust

Questions raised by tech visions

- How to enable people to access and interact with information in their everyday lives
- How to design user experiences where there is no obvious user control
- How and in what form to provide contextually-relevant information to people
- How to ensure that information passed around interconnected devices and objects is secure

Theory

- Explanation of a phenomenon
  - For example, information processing that explains how the mind, or some aspect of it, is assumed to work
- Can help identify factors relevant to the design and evaluation of interactive products
  - Such as cognitive, social, and affective
- Can be used to predict what users will do with different interfaces

Models

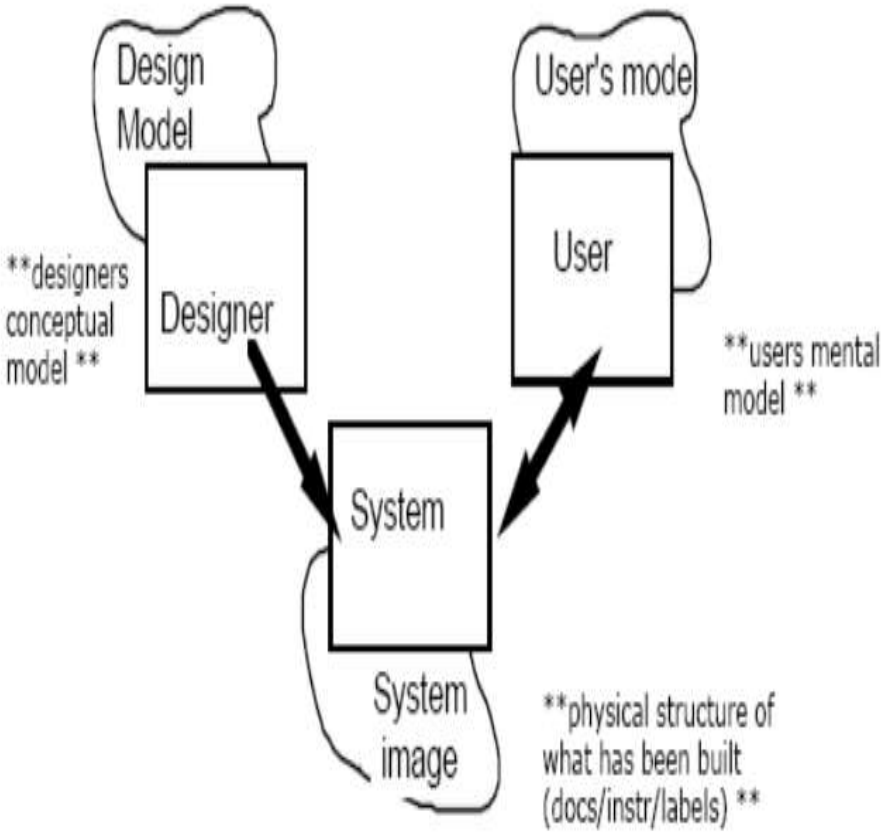
- A simplification of an HCI phenomenon
  - Enables designers to predict and evaluate alternative designs
  - Abstracted from a theory coming from a contributing discipline, for example:
    - Don Norman's (1996) model of the Seven Stages of Action:
      - User establishes the goal
      - Formulates intention
      - Specifies actions at interface
      - Executes action
      - Perceives system state
      - Interprets system state
      - Evaluates system state with respect to goal
      - Norman's model concentrates on user's view of the interface
    - Marc Hassenzahl's (2010) model of the user experience

Frameworks

- Set of interrelated concepts and-or specific questions for ‘what to look for’
- Provide advice on how to design user experiences
  - Helping designers think about how to conceptualize learning, working, socializing, fun, and emotion
- Focus on how to design particular kinds of interfaces to evoke certain responses
- Come in various forms:
  - Such as steps, questions, concepts, challenges, principles, tactics, and dimensions

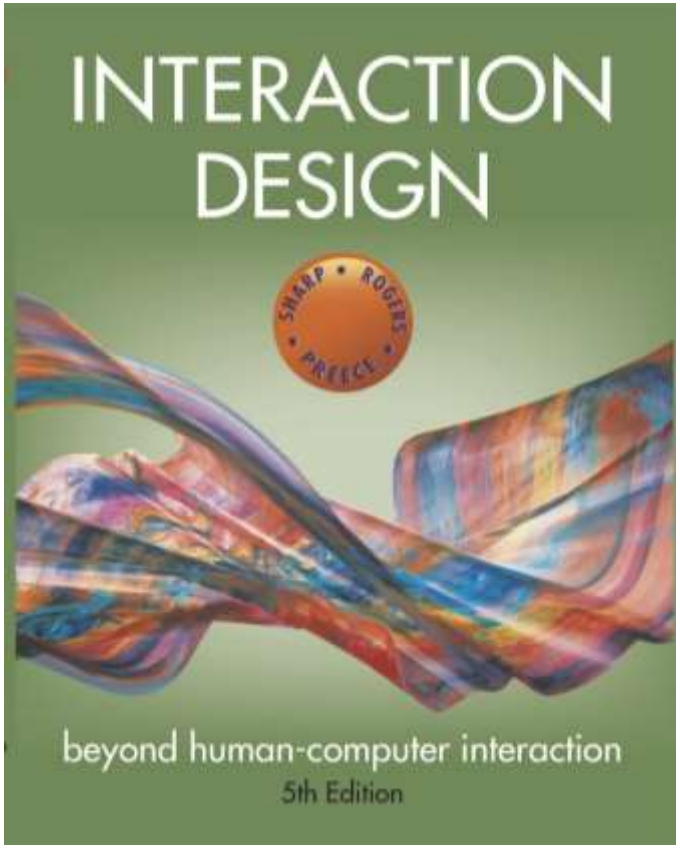
A classic HCI framework

- Don Norman's (1988) framework of the relationship between the design of a conceptual model and a user's understanding of it
- **Consists of three interacting components:**
- The Designer's Model
  - The model the designer has of how the system should work
- System Image
  - How the system actually works, which is portrayed to the user through the interface, manuals, help facilities, and so on
- The User's Model
  - How the user understands how the system works



Summary

- Developing a conceptual model involves:
  - Understanding the problem space
  - Being clear about your assumptions and claims
  - Specifying how the proposed design will support users
- A conceptual model is a high-level description of a product in terms of:
  - What users can do with it and the concepts they need to understand how to interact with it
- Interaction types provide a way of thinking about how to support user's activities
- Paradigms, visions, theories, models, and frameworks
  - Provide ways of framing design and research



Chapter 4

COGNITIVE ASPECTS

Learning Objectives

- **The main goals of this chapter are to accomplish the following:**

- Explain what cognition is and why it is important for interaction design.
- Discuss what attention is and its effects on our ability to multitask.
- Describe how memory can be enhanced through technology aids.
- Show the difference between various cognitive frameworks that have been applied to HCI.
- Explain what are mental models.
- Enable you to elicit a mental model and understand what it means.

What is cognition?

- Thinking, remembering, learning, daydreaming, decision-making, seeing, reading, talking, writing…
- Ways of classifying cognition at a higher level:
  - Experiential vs. reflective cognition (Norman, 1993)
  - Fast vs slow thinking (Kahneman, 2011)

What is cognition?

- **Experiential vs. reflective cognition (Norman, 1993):**
- **Experiential cognition** is a state of mind where people perceive, act, and react to events around them intuitively and effortlessly. It requires reaching a certain level of expertise and engagement.
  - **Examples:** Driving a car, reading a book, having a conversation, and watching a video.
- **Reflective cognition** involves mental effort, attention, judgment, and decision-making, which can lead to new ideas and creativity.
  - **Examples:** Designing, learning, and writing a report.
- Both modes are essential for everyday life.

Which involves fast vs slow thinking?

- **Fast vs slow thinking (Kahneman, 2011)**
- **Fast thinking** is similar to Don Norman's **experiential mode** insofar as it is instinctive, reflexive, and effortless, and it has no sense of voluntary control.
- **Slow thinking** takes more time and is considered to be more logical and demanding, and it requires greater concentration.
- The difference between the two modes is easy to see when asking someone to give answers to the following two arithmetic equations:
  - 2 + 2 =
  - 21 × 29 =

Which involves fast vs slow thinking?

- **Fast vs slow thinking (Kahneman, 2011)**
- What color eyes do you have?
- How many colors are there in the rainbow?
- How many months in the year have 31 days?
- What is the name of the first school you attended?

How can understanding cognition help?

- Provides knowledge about what users can and cannot be expected to do
- Identifies and explains the nature and causes of problems that users encounter
- Provides theories, modeling tools, guidance, and methods that can lead to the design of better interactive products

Cognitive processes

- Other ways of describing cognition are in terms of the context in which it takes place, the tools that are employed, the artifacts and interfaces that are used, and the people involved (Rogers, 2012).
- Depending on when, where, and how it happens, cognition can be distributed, situated, extended, and embodied.
- Cognition has also been described in terms of specific kinds of processes (Eysenck and Brysbaert, 2018). These include the following:
  - Attention
  - Perception
  - Memory
  - Learning
  - Reading, speaking and listening
  - Problem-solving, planning, reasoning and decision-making

Attention

- **Attention** is central to everyday life. It enables us to cross the road without being hit by a car or bicycle, notice when someone is calling our name, and be able to text while at the same time watching TV.
- It involves selecting things on which to concentrate, at a point in time, from the range of possibilities available, allowing us to focus on information that is relevant to what we are doing.
- The extent to which this process is easy or difficult depends on (1) whether someone has clear goals and (2) whether the information they need is salient in the environment.

Attention

- Allows us to focus on information that is relevant to what we are doing
- Involves audio and/or visual senses
- Focused and divided attention
  - Enables us to be selective in terms of the mass of competing stimuli, but limits our ability to keep track of all events
- Design recommendation
  - Information at the interface should be structured to capture users' attention, for example, use perceptual boundaries (windows), color, reverse video, sound, and flashing lights

Activity: Find the price for a double room at the Quality Inn in Pennsylvania

Activity: Find the price of a double room at the Holiday Inn in Columbia

South Carolina

City	Motel/Hotel	Area code	Phone	Rates Single	Rates Double
Charleston	Best Western	803	747-0961	$126	$130
Charleston	Days Inn	803	881-1000	$118	$124
Charleston	Holiday Inn N	803	744-1621	$136	$146
Charleston	Holiday Inn SW	803	556-7100	$133	$147
Charleston	Howard Johnsons	803	524-4148	$131	$136
Charleston	Ramada Inn	803	774-8281	$133	$140
Charleston	Sheraton Inn	803	744-2401	$134	$142
Columbia	Best Western	803	796-9400	$129	$134
Columbia	Carolina Inn	803	799-8200	$142	$148
Columbia	Days Inn	803	736-0000	$123	$127
Columbia	Holiday Inn NW	803	794-9440	$132	$139
Columbia	Howard Johnsons	803	772-7200	$125	$127
Columbia	Quality Inn	803	772-0270	$134	$141
Columbia	Ramada Inn	803	796-2700	$136	$144
Columbia	Vagabond Inn	803	796-6240	$127	$130

Activity

- Tullis (1987) found that the two screens produced quite different results
  - 1st screen: Took an average of 5.5 seconds to search
  - 2nd screen: Took 3.2 seconds to search
- Why, since both displays have the same density of information (31percent)?
- Spacing
  - In the 1st screen, the information is bunched up together, making it hard to search
  - In the 2nd screen, the characters are grouped into vertical categories of information making it easier

Multitasking and attention

- Is it possible to perform multiple tasks without one or more of them being detrimentally affected?
- Multitasking can cause people to lose their train of thought, make errors, and need to start over
- Ophir et al. (2009) compared heavy vs light multitaskers
  - Heavy multitaskers were more prone to being distracted than those who infrequently multitask
  - Heavy multitaskers are easily distracted and find it difficult to filter irrelevant information

Multitasking experiment

- Lotteridge et al. (2015) conducted another study involving writing an essay under two conditions: relevant or irrelevant information
  - Heavy multitaskers were easily distracted but able to put this to good use if the distracting sources were relevant to the task in hand
  - Irrelevant information was found to impact task performance negatively

Multitasking at work

It is increasingly common for workers to multitask

- For example, hospital workers have to attend to multiple screens in an operating room that provide new kinds of real-time information
- This requires clinician's constant attention to check if any data is unusual or anomalous
- Need to develop new attention and scanning strategies

Is it OK to use a phone when driving?

No!

- Driving is very demanding
- Drivers are prone to being distracted
- There is a significant chance of causing accidents
- Drivers' reaction times are longer to external events when talking on the phone in a car (Caird et al., 2018)
- Drivers using their phones rely more on their expectations about what is likely to happen next as conducting a conversation takes up their attention
- Response time is slower to unexpected events (Briggs et al., 2018)
- Drivers often try to imagine what the other person's face is like−the person to whom they are speaking
  - Doing so competes with the processing resources needed to enable them to notice and react to what is in front of them

Are hands-free phones safer to use when driving?

- No, as same type of cognitive processing is happening when talking
- The same thing happens when talking with front seat passenger
  - But both can stop in mid-sentence if a hazard is spotted allowing the driver to switch immediately to the road
  - So, it's less dangerous talking to a front seat passenger than a remote person
  - A remote person on the end of a phone is not privy to what the driver is seeing and will carry on the conversation when there is a hazard
  - This makes it difficult for the driver to switch all their attention to the road

Design implications for attention

- Context: Make information salient when it needs to be attended to at a given stage of a task
- Use techniques to achieve this:
  - For example, color, ordering, spacing, underlining, sequencing, and animation
- Avoid cluttering visual interfaces with too much information
- Consider designing different ways to support effective switching and returning to an interface

Perception

- How information is acquired from the world and transformed into experiences
- Obvious implication is to design representations that are readily perceivable, for instance:
  - Text should be legible
  - Icons should be easy to distinguish and read

Is color contrast good? Find Italian

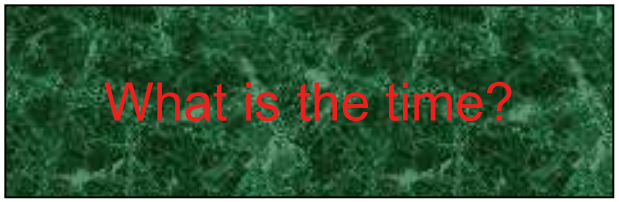
Black Hills Forest	Peters Landing	Jefferson Farms	Devlin Hall
Cheyenne River	Public Health	Psychophysics	Positions
Social Science	San Bernardino	Political Science	Hubard Hall
South San Jose	Moreno Valley	Game Schedule	Fernadino Beach
Badlands Park	Altamonte Springs	South Addision	Council Bluffs
Juvenile Justice	Peach Tree City	Cherry Hills Village	Classical Lit
Results and Stats	Highland Park	Creative Writing	Sociology
Thousand Oaks	Manchesney Park	Lake Havasu City	Greek
Promotions	Vallecito Mts.	Engineering Bldg	Wallace Hall
North Palermo	Rock Falls	Sports Studies	Concert Tickets
Credit Union	Freeport	Lakewood Village	Public Radio FM
Wilner Hall	Slaughter Beach	Rock Island	Children's Museum
Performing Arts	Rocky Mountains	Deerfield Beach	Writing Center
Italian	Latin	Arlington Hill	Theater Auditions
Coaches	Pleasant Hills	Preview Game	Delaware City
McKees Rocks	Observatory	Richland Hills	Scholarships
Glenwood Springs	Public Affairs	Experts Guide	Hendricksville
Urban Affairs	Heskett Center	Neff Hall	Knights Landing
McLeansboro	Brunswick	Grand Wash Cliffs	Modern Literature
Experimental Links	East Millinocket	Indian Well Valley	Studio Arts
Graduation	Women's Studies	Online Courses	Hughes Complex
Emory Lindquist	Vacant	Lindquist Hall	Cumberland Flats
Clinton Hall	News Theatre	Fisk Hall	Central Village
San Luis Obispo	Candlewood Isle	Los Padres Forest	Hoffman Estates

Are borders and white space better? Find French

Activity

- Weller (2004) found people took less time to locate items for information that was grouped
  - Using a border (2nd screen) compared with using color contrast (1st screen)
- Some argue that too much white space on web pages is detrimental to search process
  - Makes it hard to find information
- Do you agree?

Activity: Which is the easiest to read and why?



What is the time?

What is the time?

What is the time?

What is the time?

Design implications

- Icons should enable users to *distinguish* their meaning readily
- Bordering and spacing are effective visual ways of grouping information
- Sounds should be audible and distinguishable
- Research proper color contrast techniques when designing an interface:
  - Yellow on black or blue is fine
  - Yellow on green or white is a no-no
- Haptic feedback should be used judiciously

Memory

- Involves recalling various kinds of knowledge that allow people to act appropriately
  - For example, recognizing someone's face or remembering someone's name
- First encode and then retrieve knowledge
- We don't remember everything−it involves filtering and processing what is attended to
- Context is important as to how we remember (that is, where, when, how, and so on)
- We recognize things much better than being able to recall things
- We remember less about objects that we have photographed than when we observe them with the naked eye (Henkel, 2014)

Processing in memory

- Encoding is first stage of memory
  - Determines which information is attended to in the environment and how it is interpreted
- The more attention paid to something…
- The more it is processed in terms of thinking about it and comparing it with other knowledge…
- The more likely it is to be remembered
  - For example, when learning about HCI, it is much better to reflect upon it, carry out exercises, have discussions with others about it, and write notes than just passively read a book, listen to a lecture or watch a video about it

Context is important

- Context affects the extent to which information can be subsequently retrieved
- Sometimes it can be difficult for people to recall information that was encoded in a different context:
  - “You are on a train and someone comes up to you and says hello. You don’t recognize him for a few moments, but then realize it is one of your neighbors. You are only used to seeing your neighbor in the hallway of your apartment building, and seeing him out of context makes him difficult to recognize initially”

Activity

- Try to remember the dates of your grandparents' birthday
- Try to remember the cover of the last two books you read
- Which was easiest? Why?
- People are very good at remembering visual cues about things
  - For instance, the color of items, the location of objects and marks on an object
- They find it more difficult to learn and remember arbitrary material
  - For example, birthdays and phone numbers

Recognition versus recall

- Command-based interfaces require users to recall from memory a name from a possible set of 100s of names
- Graphical interfaces provide visually-based options (menus, icons) that users need only browse through until they recognize one
- Web browsers provide tabs and history lists of visited URLs that support recognition memory

The problem with the classic ‘7,+ or − 2’

- George Miller’s (1956) theory of how much information people can remember
- People’s immediate memory capacity is very limited to 7, + or − 2
- Has been applied in interaction design when considering how many options to display
- But is it a good use of a theory in HCI?
- Is it helpful?

When creating an interface, should the designer…

- Present only 7 options on a menu
- Display only 7 icons on a tool bar
- Have no more than 7 bullets in a list
- Place only 7 items on a pull down menu
- Place only 7 tabs on the top of a website page?
- Not necessarily…

The reason is…

- People can scan lists of bullets, tabs, and menu items for the one they want
- They don’t have to recall them from memory, having only briefly heard or seen them
- So you can have more than nine at the interface
  - For instance, history lists of websites visited
- Sometimes a small number of items is good
  - For example, smart watch displays
- Depends on task and available screen estate

Personal Information management

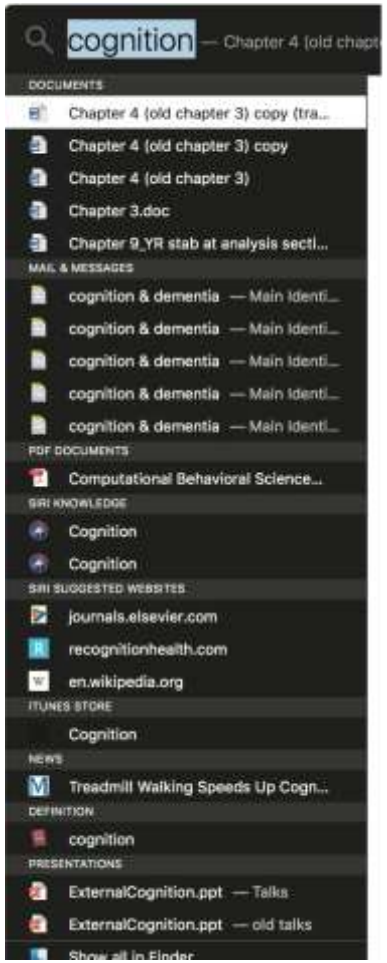
Is a growing problem for many users:

- They accumulate a vast numbers of documents, images, music files, video clips, emails, attachments, bookmarks, and so forth
- Where and how to save them all; then remembering what they were called and where to find them again
- Naming most common means of encoding them
- But can be difficult to remember, especially when you have 10,000s
- How might such a process be facilitated taking into account people's memory abilities?

Personal Information management

- Bergman and Whittaker, three interdependent processes model (2016) to help people manage their stuff:
  I. How to decide what stuff to keep
  II. How to organize it when storing
  III. Which strategies to use to retrieve it later
- Most common approach is to use folders and naming
- Strong preference for scanning across and within folders when looking for something
- Search engines only helpful if you know the name of the file
- Smart search engines help with listing relevant files for partial name or when type in first letter

Apple’s Spotlight search tool



Memory load

- Online/mobile and phone banking now require users to provide multiple pieces of information to access their account
  - For instance, ZIP code, birthplace, a memorable date, first school attended
  - Known as multifactor authentication (MFA)
- Why?
  - Increased security concerns
- Password managers, such as LastPass, have been developed that require only one master password
  - Reduces stress and memory load on users
- Passwords could become extinct with the widespread use of biometrics and computer vision algorithms

Digital Forgetting

- When might you wish to forget something that is online?
  - When you break up with a partner
  - Emotionally painful to be reminded of them through shared photos, social media, and so on.
- Sas and Whittaker (2013) suggest ways of harvesting and deleting digital content
  - For example, making photos of ex into an abstract collage
  - Helps with closure

Memory aids

- SenseCam, developed by Microsoft Research Labs (now Autographer)
  - A wearable device that intermittently takes photos without any user intervention while worn
  - Digital images taken are stored and revisited using special software
  - Has been found to improve people's memory, especially those suffering from dementia
- Other aids include RemArc, which triggers long-term memory using old BBC materials

SenseCam

Design implications

- Reduce cognitive load by avoiding long and complicated procedures for carrying out tasks
- Design interfaces that promote recognition rather than recall
- Provide users with various ways of labelling digital information to help them easily identify it again
  - For example, folders, categories, color, flagging, and time stamping

Learning

- Involves the accumulation of skills and knowledge involving memory
- Two main types:
  - Incidental learning (for example, recognizing people's faces, what you did today)
  - Intentional learning (for instance, studying for an exam, learning to cook)
  - Intentional learning is much harder!
  - Many technologies have been developed to help (for example, multimedia, animations, VR)
- People find it hard to learn by following instructions in a manual
- People prefer to learn by doing

Design implications

- Design interfaces that encourage exploration
- Design interfaces that constrain and guide learners
- Dynamically linking concepts and representations can facilitate the learning of complex material

Reading, speaking, and listening

The ease with which people can read, listen, or speak differs:

- Many prefer listening to reading
- Reading can be quicker than speaking or listening
- Listening requires less cognitive effort than reading or speaking
- Dyslexics have difficulties understanding and recognizing written words

Applications

- Voice user interfaces allow users to interact with them by asking questions
  - For example, Google Voice, Siri, and Alexa
- Speech-output systems use artificially-generated speech
  - For instance, written text-to-speech systems for the visually impaired
- Natural-language systems enable users to type in questions and give text-based responses
  - Such as, chatbots

Design implications

- Speech-based menus and instructions should be short
- Accentuate the intonation of artificially generated speech voices
  - They are harder to understand than human voices
- Provide opportunities for making text large on a screen

Problem-solving, planning, reasoning, and decision-making

- All these processes involve *reflective* cognition
  - For example, thinking about what to do, what the options are, and the consequences
- Often involves conscious processes, discussion with others (or oneself), and the use of artifacts
  - Such as maps, books, pen and paper
- May involve working through different scenarios and deciding which is best option
- Weighing up alternatives

Design implications

- Provide information and help pages that are easy to access for people who wish to understand more about how to carry out an activity more effectively (for example, web searching)
- Use simple and memorable functions to support rapid decision-making and planning

Dilemma

- The app mentality is making it worse for people to make their own decisions because they are becoming risk averse (Gardner and Davis, 2013)
  - Instead, they now rely on a multitude of apps
  - This makes them increasingly anxious
  - They are unable to make decisions by themselves
  - They need to resort to looking up info, getting other's opinions on social media, and comparing notes
- Do you agree?
- Did it happen to you when deciding which university/school to attend?

Cognitive frameworks

- These are used to explain and predict user behavior at the interface
  - Based on theories of behavior
  - Focus is on mental processes that take place
  - Also use of artifacts and representations
- Most well known are:
  - Mental models
  - Gulfs of execution and evaluation
  - Distributed cognition
  - External and embodied cognition

Mental models

- Users develop an understanding of a system through learning about and using it
- Knowledge is sometimes described as a mental model:
  - How to use the system (what to do next)
  - What to do with unfamiliar systems or unexpected situations (how the system works)
- People make inferences using mental models of how to carry out tasks

More mental models

- Craik (1943) described mental models as:
  - Internal constructions of some aspect of the external world enabling predictions to be made
- Involves unconscious and conscious processes
  - Imagery and analogies are activated
- Deep versus shallow models
  - For example, how to drive a car and how it works

Everyday reasoning and mental models

(a) You arrive home on a cold winter's night to a cold house. How do you get the house to warm up as quickly as possible? Set the thermostat to be at its highest or to the desired temperature?

(b) You arrive home starving hungry. You look in the fridge and find all that is left is an uncooked pizza. You have an electric oven. Do you warm it up to 375 degrees first and then put it in (as specified by the instructions) or turn the oven up higher to try to warm it up quicker?

Heating up a room or oven that is thermostat-controlled

- Many people when asked (a) choose the first option
- Why?
  - They think it will heat the room up quicker
  - General valve theory, where ‘more is more’ principle is generalized to different settings (for instance, gas pedal, gas cooker, tap, radio volume)
  - But it is a wrong mental model for thermostats based on on-off switch model
- Many people when asked (b) choose the first option
  - Electric ovens work on the same principle as thermostats
- Most of us have erroneous mental models (Kempton, 1996)

Erroneous mental models

- Lots of people hit the button for elevators and pedestrian crossings at least twice
  - Why? Think it will make the lights change faster or ensure that the elevator arrives!
- What kinds of mental models do users have for understanding how interactive devices work?
  - Poor, often incomplete, easily confusable, based on inappropriate analogies and superstition (Norman, 1983)

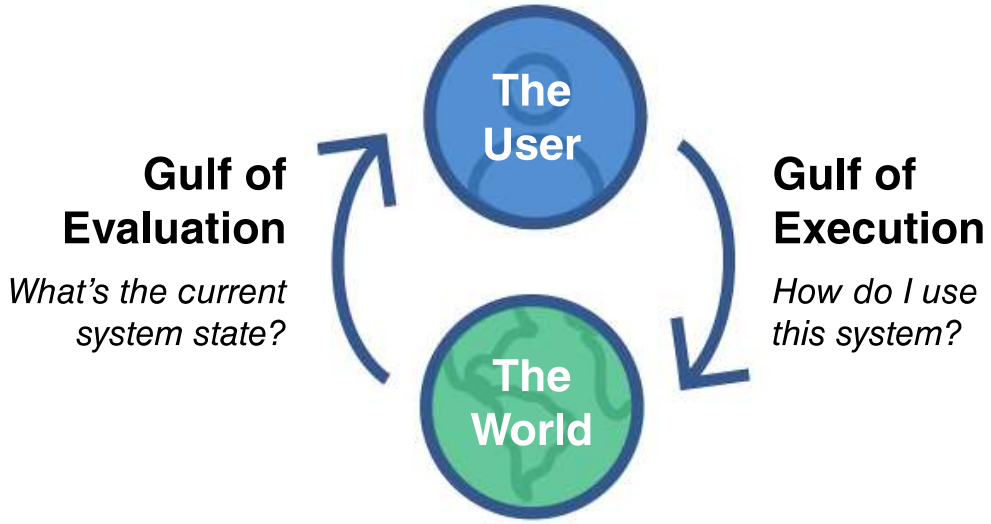
How can UX be designed to help people build better mental models?

- Clear and easy to use instructions
- Appropriate tutorials and contextual sensitive guidance
- Provide online videos and chatbot windows when needing help
- Transparency: to make interfaces intuitive to use
- Affordances of what actions an interface allows
  - For example, swiping, clicking, or selecting

Gulfs of execution and evaluation

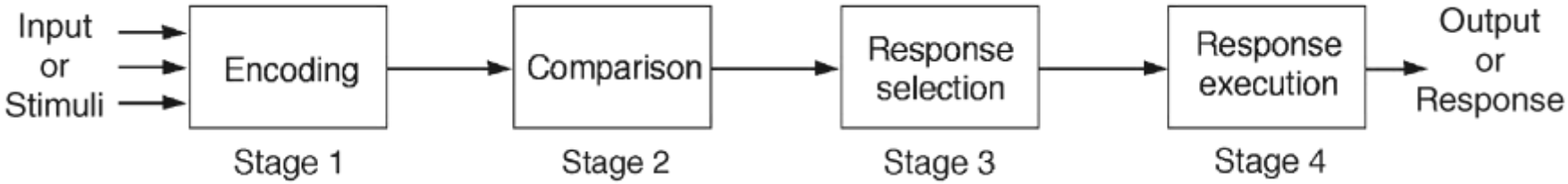
- The ‘gulfs’ explicate the gaps that exist between the user and the interface
- The gulf of execution
  - The distance from the user to the physical system
- The gulf of evaluation
  - The distance from the physical system to the user
- Bridging the gulfs can reduce cognitive effort required to perform tasks
- Can reveal whether interface increases or decreases cognitive load and whether it is obvious what to do next (Norman, 1986; Hutchins et al, 1986)

Bridging the gulfs



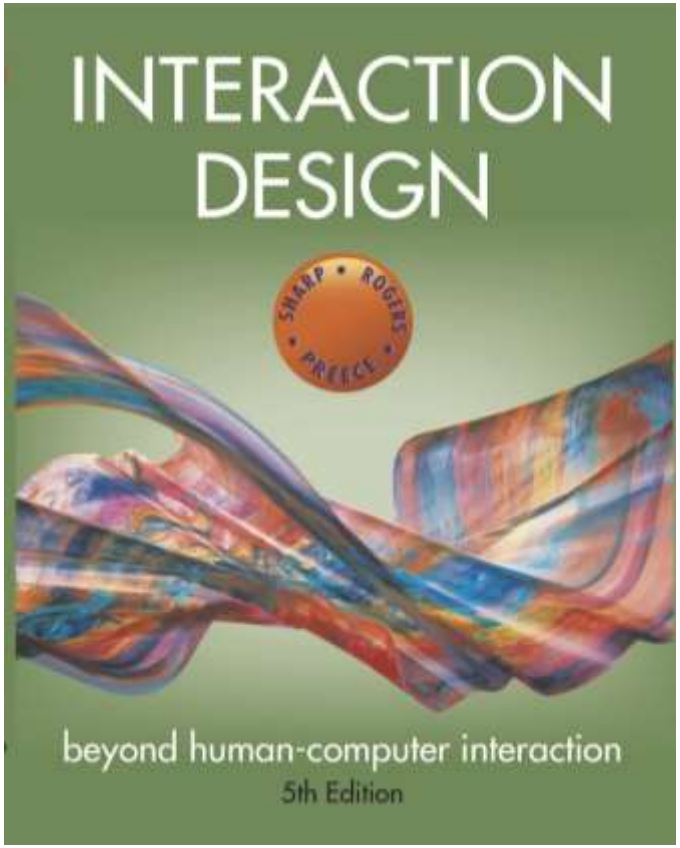
Information processing

- Conceptualizes human performance in metaphorical terms of information processing stages



Limitations

- Based on modeling mental activities that happen exclusively inside the head
- Do not adequately account for how people interact with computers and other devices in real world



Chapter 8

DATA GATHERING

Five key issues

1. Setting goals
   - Decide how to analyze data once collected
2. Identifying participants
   - Decide from whom to gather data
   - How many participants are needed
3. Relationship with participants
   - Clear and professional
   - Informed consent when appropriate
   - Triangulation
   - Look at data from more than one perspective
   - Collect more than one type of data, for instance, qualitative data from experiments and qualitative data from interviews
4. Pilot studies
   - Small trial of main study

Data recording

- Notes, audio, video, and photographs can be used individually or in combination:
  - Notes plus photographs
  - Audio plus photographs
  - Video
- Different challenges and advantages with each type of data recording

Interviews

Unstructured: Not directed by a script. Rich but not replicable.

Structured: Tightly scripted, often like a questionnaire. Replicable but may lack richness.

Semi-structured: Guided by a script, but interesting issues can be explored in more depth. Can provide a good balance between richness and replicability.

Focus groups: A group interview

Interview questions

- Two types:
  - ‘Closed questions’ have a predetermined answer format, for example, ‘yes’ or ‘no’
  - ‘Open questions’ do not have a predetermined format
- Closed questions are easier to analyze
- Avoid:
  - Long questions
  - Compound sentences — split them into two
  - Jargon and language that the interviewee may not understand
  - Leading questions that make assumptions, for example, why do you like …?

Running the interview

Introduction: Introduce yourself, explain the goals of the interview, reassure about the ethical issues, ask to record, and present the informed consent form.

Warm-up: Make first questions easy and non-threatening.

Main body: Present questions in a logical order

A cool-off period: Include a few easy questions to defuse tension at the end

Closure: Thank interviewee, signal the end, for example, switch recorder off.

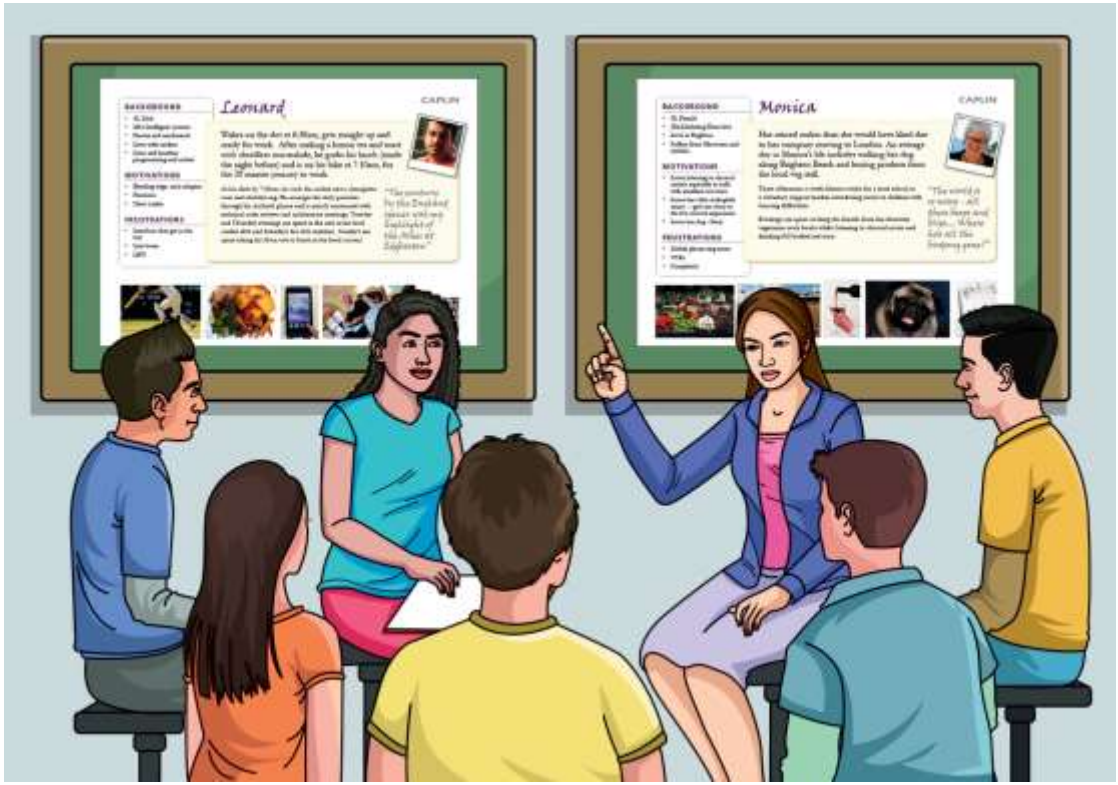
Other forms of interviews

Digital conferencing systems such as Skype, Zoom, email, and smartphones can be used to conduct interviews. Some advantages are:

- Participants are in their own environment so are more relaxed
- Participants don’t need to travel
- Participants don’t need to worry about what to wear
- For interviews involving sensitive issues, it is easier for interviewees to be anonymous

Enriching the interview process

Props: Devices for prompting interviewee, for example, use a prototype, scenario



Questionnaires

- Questions can be closed or open
- Closed questions are easier to analyze, and may be distributed and analyzed by computer
- They can be administered to large populations
- Disseminated by paper, email and the web
- Sampling can be a problem when the size of a population is unknown as is common online evaluation

Questionnaire design

- The impact of a question can be influenced by question order.
- You may need different versions of the questionnaire for different populations.
- Provide clear instructions on how to complete the questionnaire.
- Strike a balance between using white space and keeping the questionnaire compact.
- Avoid very long questions and questionnaires
- Decide on whether phrases will all be positive, all negative, or mixed.

Question and response format

- ‘Yes’ and ‘No’ checkboxes
- Checkboxes that offer many options
- Rating scales
  - Likert scales
  - Semantic scales
  - 3, 5, 7 or more points
- Open-ended responses

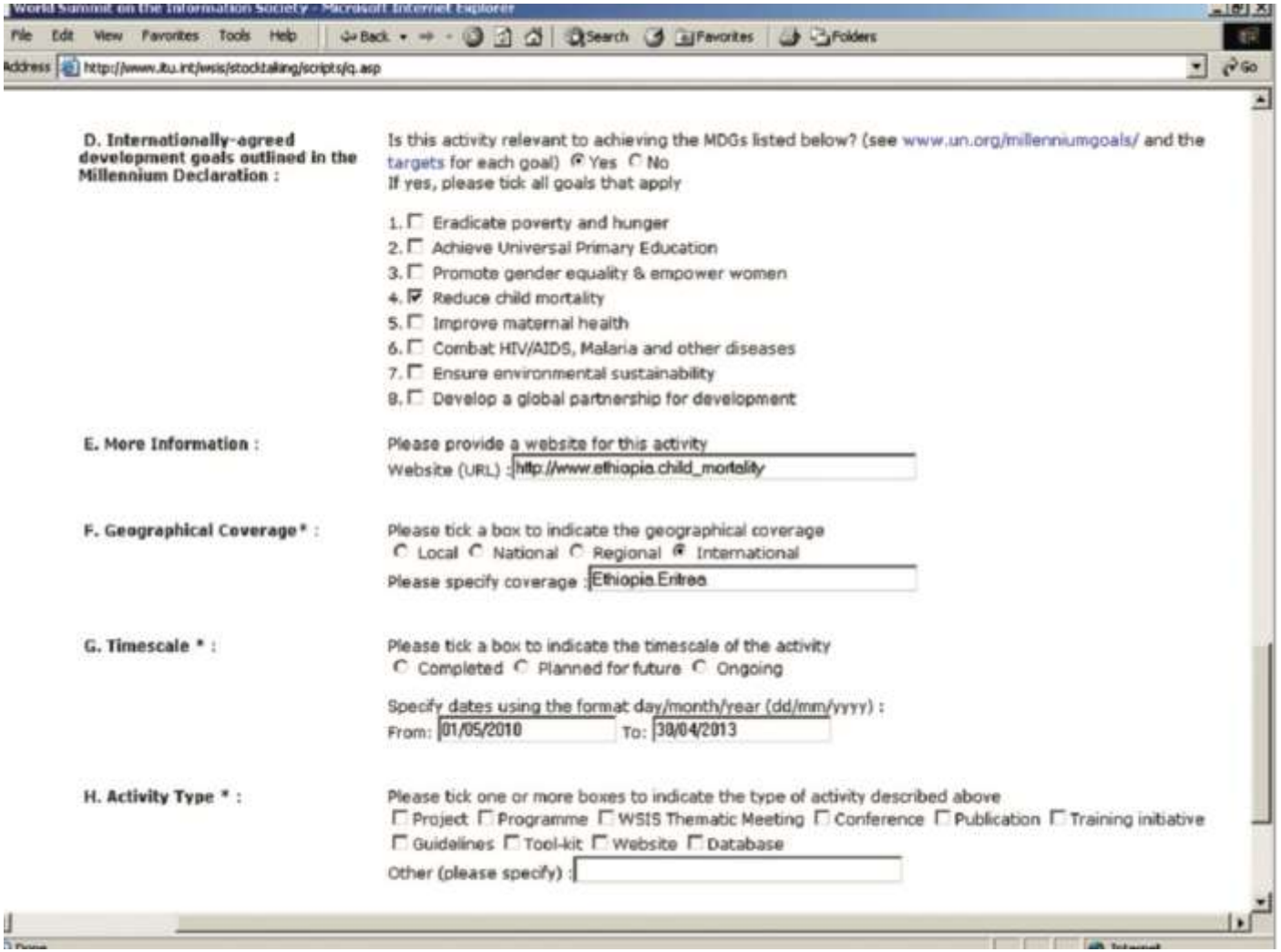
Encouraging a good response

- Make sure that the purpose of study is clear
- Promise anonymity
- Ensure that questionnaire is well designed
- Offer a short version for those who do not have time to complete a long questionnaire
- If mailed, include a stamped, addressed envelope
- Follow-up with emails, phone calls, or letters
- Provide an incentive
- 40 percent response rate is good, 20 percent is often acceptable

Advantages of online questionnaires

- Relatively easy and quick to distribute
- Responses are usually received quickly
- No copying and postage costs
- Data can be collected in database for analysis
- Time required for data analysis is reduced
- Errors can be corrected easily

Example of an online questionnaire



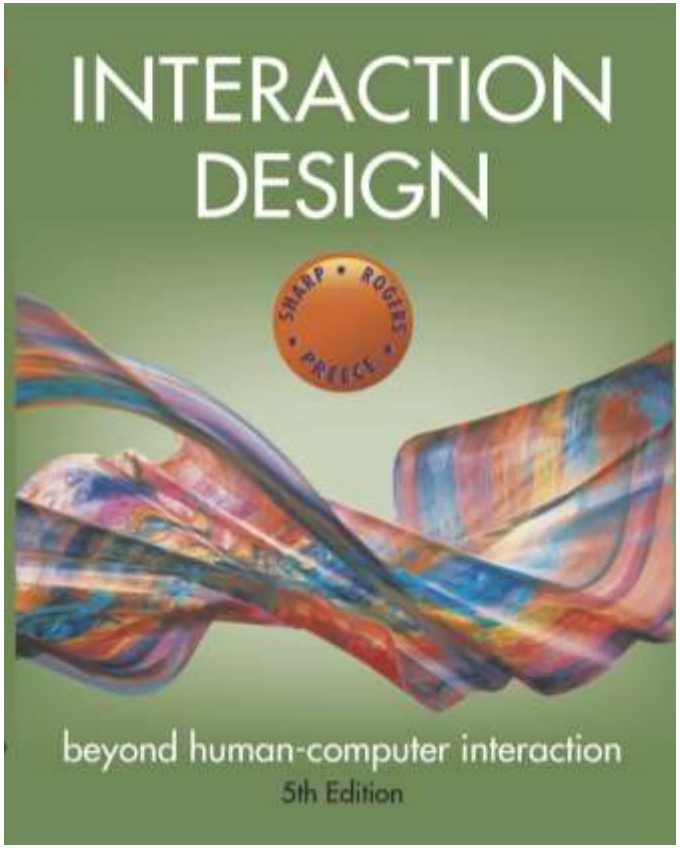
Questionnaire shows check boxes, radio boxes, and pull-down menus

Problems with online questionnaires

- Sampling is problematic if population size is unknown
- Preventing individuals from responding more than once can be a problem
- Individuals have also been known to change questions in email questionnaires

Deploying online questionnaires

- Plan the timeline
- Design offline
- Program/complete online template
- Test the survey to make sure that it behaves as you would expect
- Test it with a group that will not be part of the survey to check that the questions are clear
- Recruit participants



Chapter 11

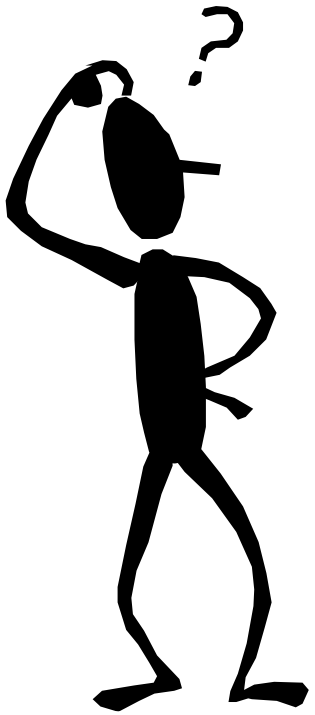
DISCOVERING REQUIREMENTS

Learning Objectives

- **The main goals of this chapter are to accomplish the following:**
- Describe different kinds of requirements.
- Allow you to identify different kinds of requirements from a simple description.
- Explain additional data gathering techniques and how they may be used to discover requirements.
- Enable you to develop a persona and a scenario from a simple description.
- Describe use cases as a way to capture interaction in detail.

Overview

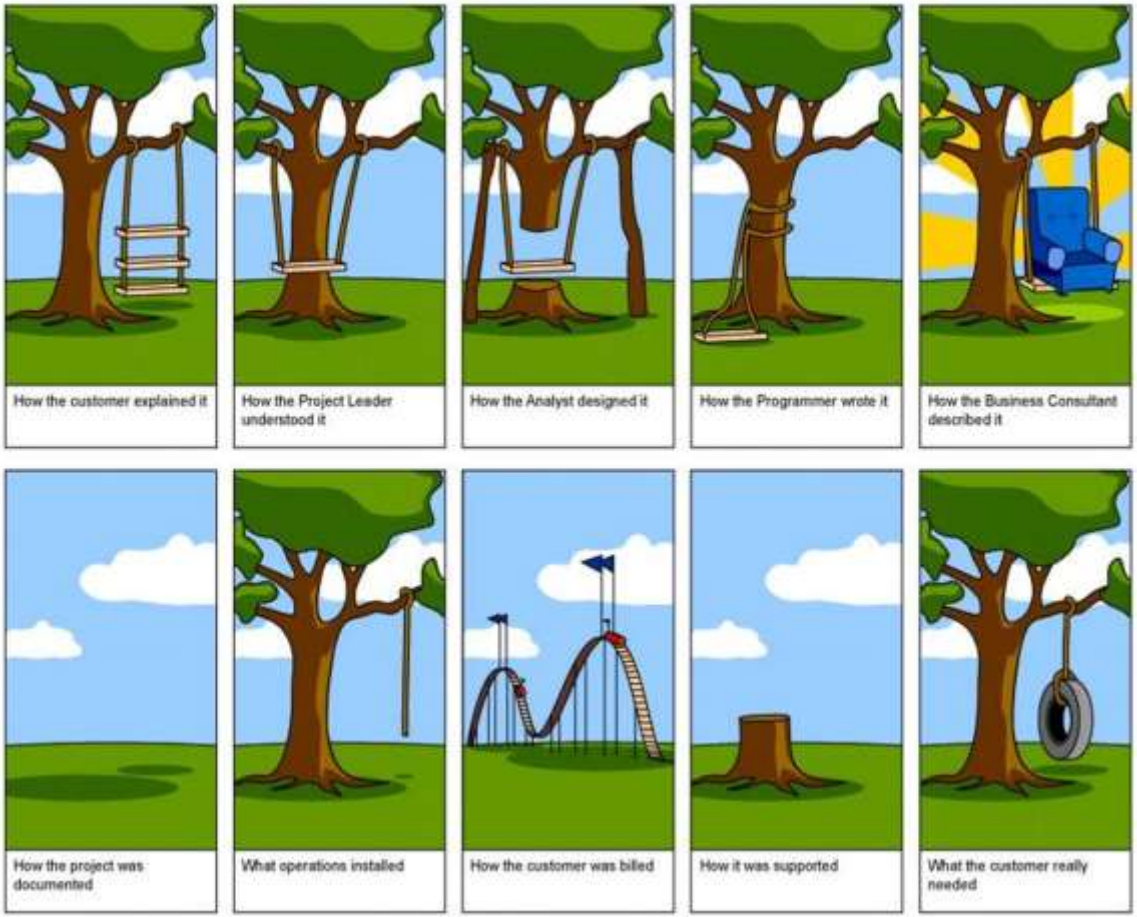
- The importance of requirements
- Different types of requirements
- Data gathering for requirements
- Bringing requirements to life
  - Personas
  - Scenarios
- Capturing interaction with user cases



What, how and why?

- What is the purpose of the requirements activity?
  - Explore the problem space
  - Establish a description of what will be developed
- How to capture requirements once discovered
  - In prototypes or operational product
  - Through structured or rigorous notations
  - Different capturing mechanisms emphasize and de-emphasize different aspects

Why bother?



Requirements activity is the stage where miscommunication occurs most commonly

What are requirements?

- A statement about an intended product that specifies what it is expected to do or how it will perform
- Different forms and different levels of abstraction
- User stories (most prevalent in agile development contexts)
- Format:

  As a <role>, I want <behavior> so that <benefit>

- Example user stories for a travel organizer might be:

  As a <traveler>, I want <to save my favorite airline for all my flights> so that <I will be able to collect air miles>

  As a <travel agent>, I want <my special discount rates to be displayed to me> so that <I can offer my clients competitive rates>

Volere shell

Requirement #: 75 Requirement Type: 9 Event/use case #: 6

Description: The product shall issue an alert if a weather station fails to transmit readings.

Rationale: Failure to transmit readings might indicate that the weather station is faulty and needs maintenance, and that the data used to predict freezing roads may be incomplete.

Source: Road Engineers
Fit Criterion: For each weather station the product shall communicate to the user when the recorded number of each type of reading per hour is not within the manufacturer's specified range of the expected number of readings per hour.

Customer Satisfaction: 3 Customer Dissatisfaction: 5

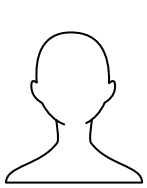
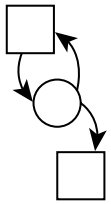
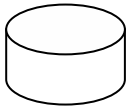
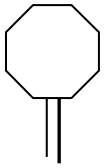
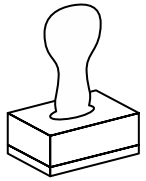
Dependencies: None Conflicts: None

Supporting Materials: Specification of Rosa Weather Station

History: Raised by GBS, 28 July

Volere

The seven product dimensions

						
User	Interface	Action	Data	Control	Environment	Quality Attribute
Users interact with the product	The product connects to users, systems, and devices	The product provides capabilities for users	The product includes a repository of data and useful information	The product enforces constraints	The product conforms to physical properties and technology platforms	The product has certain properties that qualify its operation and development

Source: Gottesdiener and Gorman (2012), p.58. Used courtesy of Ellen Gottesdiener

Different kinds of requirements

- Functional:
  - What the system should do
- Data:
  - What kinds of data need to be stored?
  - How will they be stored (for example, database)?

Different kinds of requirements

Environment or context of use:

Physical: dusty? noisy? vibration? light? heat? humidity? …. (for example, in a hospital)

Social: collaboration and co-ordination, data sharing, distributed, synchronous or asynchronous, privacy

Organizational: user support, communications structure and infrastructure, availability of training

Technical: On what technologies will it run or need to be compatible?

Different kinds of requirements

Users — Who are they?

- **Characteristics**: nationality, educational background, attitude to computers
- **System use**: novice, expert, casual, frequent

  Novice: prompted, constrained, clear

  Expert: flexibility, access/power

  Frequent: shortcuts

  Casual/infrequent: clear menu paths
- User profile

Different kinds of requirements

- Usability goals
- User experience goals
- Different products have different requirements and may be implemented in different ways, for example, trustworthiness

Usable security

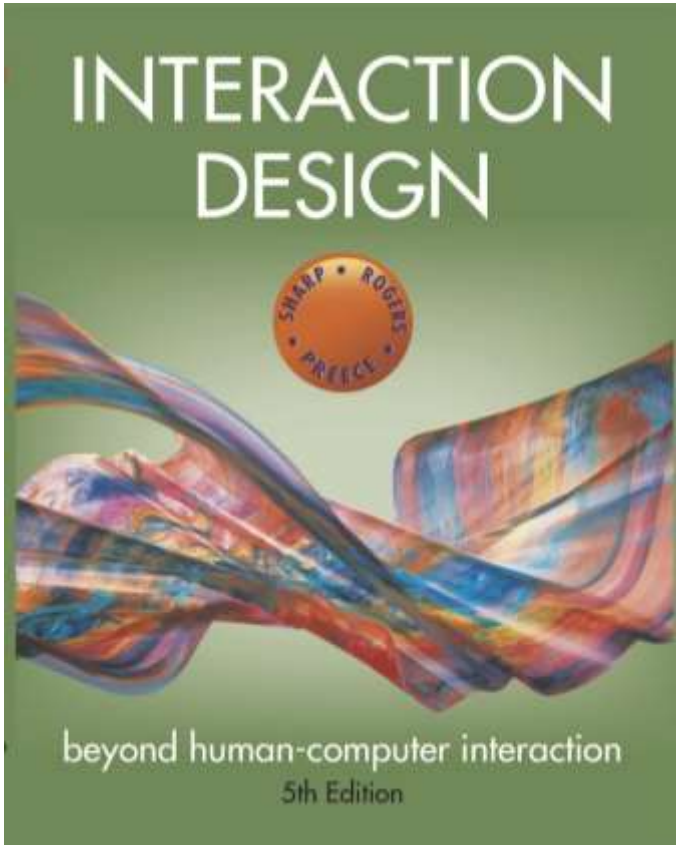
How to make security robust without detracting from user experience

- If the usability of security is ignored, then security mechanisms will be circumvented
- Passwords as an example
  - Too much advice about how to choose a password
  - Coping strategies may compromise security

Click t

Data gathering for requirements

- Interviews, observation, and questionnaires
- Studying documentation:
  - Procedures and rules are often written down in manuals
  - Good source of data about the steps involved in an activity and any regulations governing a task
  - Not to be used in isolation
  - Good for understanding legislation and getting background information
  - No stakeholder time, which is a limiting factor for other techniques
- Researching similar products:
  - Good for prompting requirements



Chapter 12

DESIGN, PROTOTYPING and CONSTRUCTION

Learning Objectives

- **The main goals of this chapter are to accomplish the following:**
- Describe prototyping and the different types of prototyping activities.
- Enable you to produce simple prototypes from the models developed during the requirements activity.
- Enable you to produce a conceptual model for a product and justify your choices.
- Explain the use of scenarios and prototypes in design.
- Introduce both physical computing kits and software development kits and their role in construction.

Overview

- Prototyping
- Conceptual design
- Concrete design
- Using scenarios
- Generating prototypes
- Construction

Prototyping

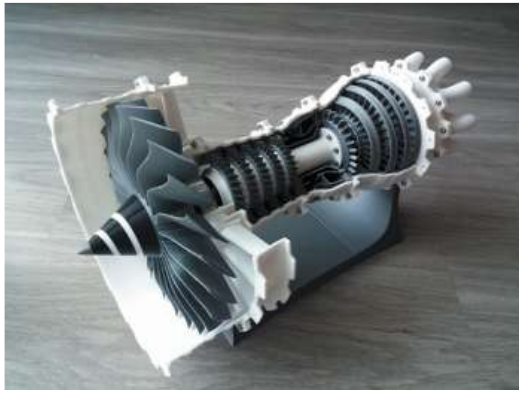
- What is a prototype?
- Why prototype?
- Different kinds of prototyping
  - Low fidelity
  - High fidelity
- Compromises in prototyping
  - Vertical
  - Horizontal
- Final product needs to be engineered

What is a prototype?

- One manifestation of a design that allows stakeholders to interact with it
- In other design fields, a prototype is a small-scale model:
  - A miniature car
  - A miniature building or town

Source: PalmPilot wooden model
© Mark Richards

3D Printing Examples



(a)

(b)

(c)

Examples of 3D printing: (a) model jet engine, (b) Spider Dress 2.0 by Anouk Wipprecht, and (c) teddy bear "printed" form a wireframe design

Sources: (a) Build Your Own Jet Engine. Licensed under CC-BY-3.0, (b) arch20, and (c) used courtesy of Scott Hudson

What is a prototype in interaction design?

In interaction design, a prototype can be (among other things):

- A series of screen sketches
- A storyboard, for example, a cartoon-like series of scenes
- A PowerPoint slide show
- A video simulating the use of a system
- A lump of wood (for instance, the PalmPilot)
- A cardboard mock-up
- A piece of software with limited functionality written in the target language or in another language

Why prototype?

- Evaluation and feedback are central to interaction design
- Stakeholders can see, hold, and interact with a prototype more easily than a document or a drawing
- Team members can communicate effectively
- Ideas can be tested out
- Prototyping encourages reflection: an important aspect of design
- Prototypes answer questions and support designers in choosing between alternatives

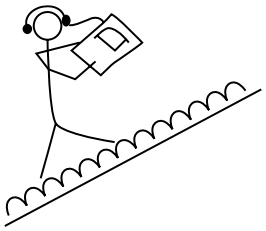
Low-fidelity Prototyping

- Uses a medium which is unlike the final medium, for example, paper or cardboard
- Is quick, cheap, and easily changed
- Examples:
  - Sketches of screens, task sequences, and so on
  - ‘Post-it’ notes
  - Storyboards
  - ‘Wizard-of-Oz’

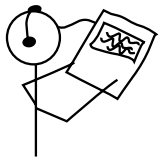
Storyboards

- It is a series of sketches showing how a user might progress through a task using the product
- Often used with scenarios, bringing in more detail and a chance to role play

Example storyboard

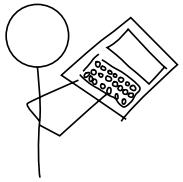


Christina walks up hill; the product gives her information about the site

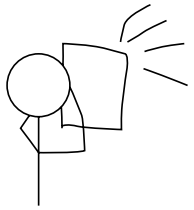


Christina adjusts the preferences to find information about the pottery trade in Ancient Greece

Christina scrambles to the highest point



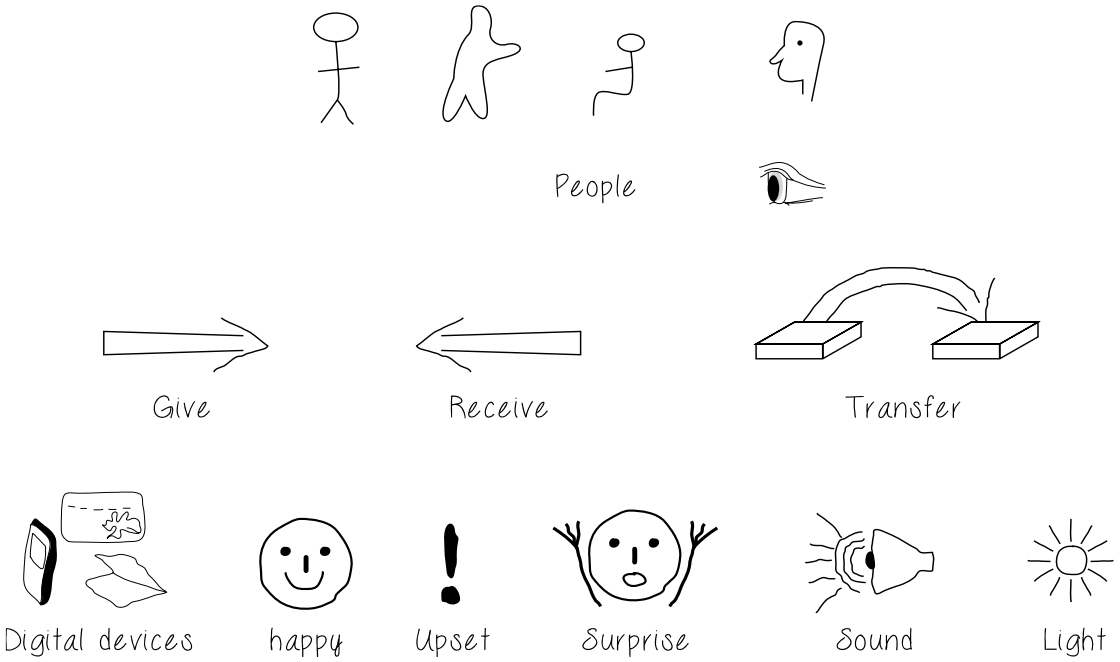
Christina stores information about the pottery trader's way of life in Ancient Greece



Christina takes a photograph of the location of the pottery market

Sketching

- Low-fidelity prototyping often relies on sketching
- Don't be inhibited about drawing ability — Practice simple symbols

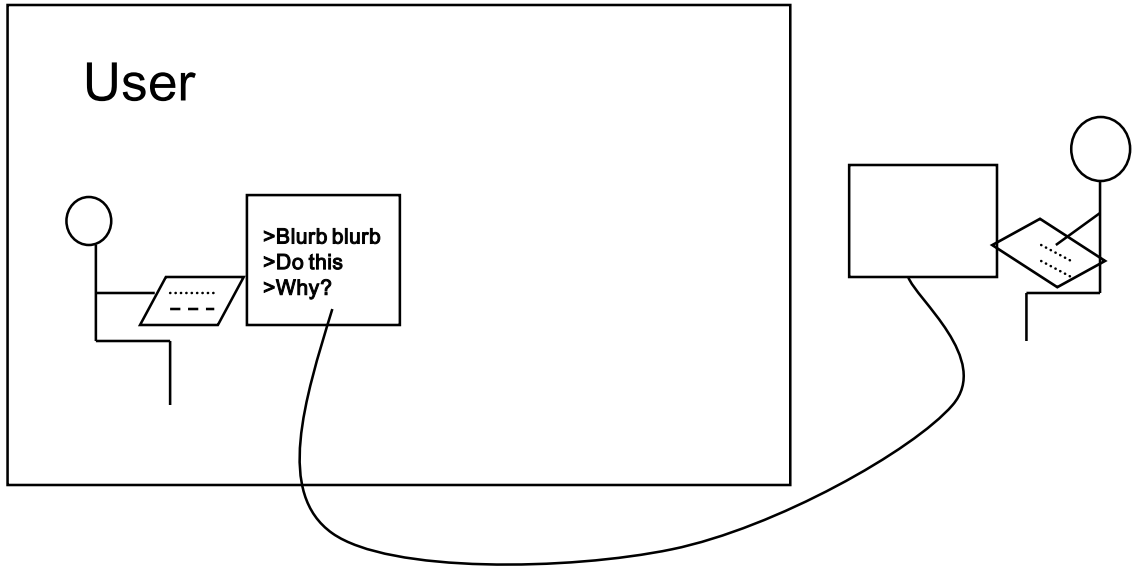


Prototyping with index cards

- Index cards (3 x 5 inches)
- Each card represents one element of interaction
- In evaluation, can step through the cards

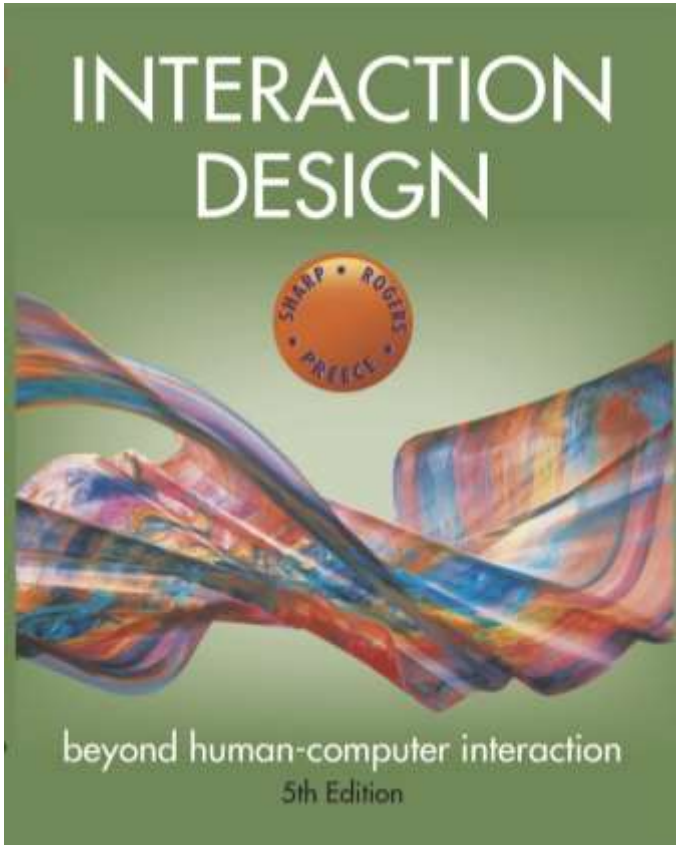
‘Wizard-of-Oz’ prototyping

- The user thinks they are interacting with a computer, but a human is responding to output rather than the system
- Usually done early in design to understand users’ expectations
- What is ‘wrong’ with this approach?



High-fidelity prototyping

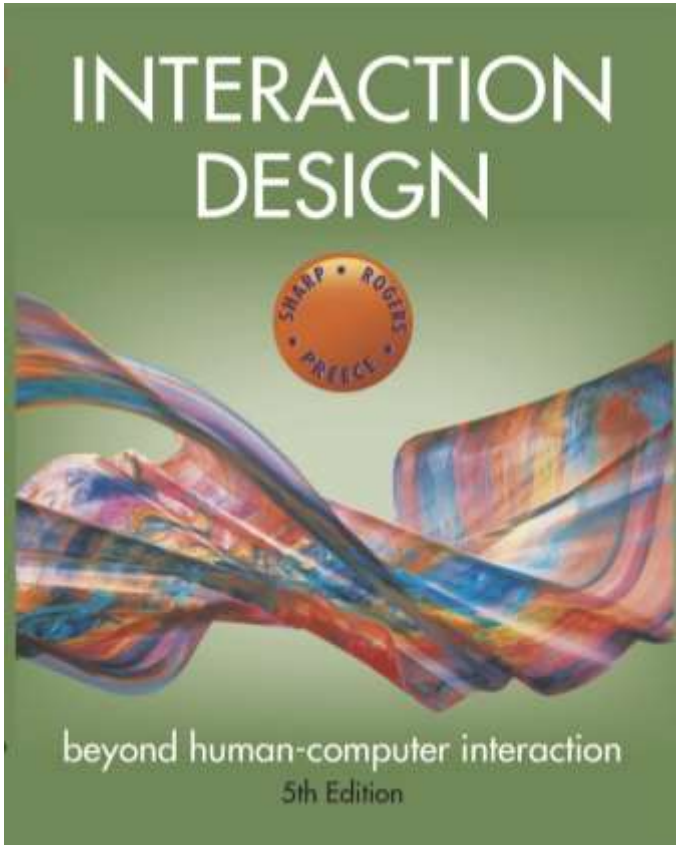
- Uses materials that you would expect to be in the final product
- Prototype looks more like the final system than a low-fidelity version
- High-fidelity prototypes can be developed by integrating existing hardware and software components
- Danger that users think they have a complete system…see compromises



Chapter 14
INTRODUCING EVALUATION

Evaluation methods

Method	Controlled settings	Natural settings	Without users
Observing	x	x	
Asking users	x	x	
Asking experts		x	x
Testing	x		
Modeling			x



Chapter 15
Evaluation Studies:
From Controlled to Natural Settings

Usability testing

- Involves recording performance of typical users doing typical tasks
- Controlled settings
- Users are observed and timed
- Data is recorded on video, and key presses are logged
- The data is used to calculate performance times and to identify and explain errors
- User satisfaction is evaluated using questionnaires and interviews
- Field observations may be used to provide contextual understanding

Quantitative performance measures

- Number of users successfully completing the task
- Time to complete task
- Time to complete task after time away from task
- Number and type of errors per task
- Number of errors per unit of time
- Number of navigations to online help or manuals
- Number of users making a particular type of error

Source: Wixon and Wilson, 1997

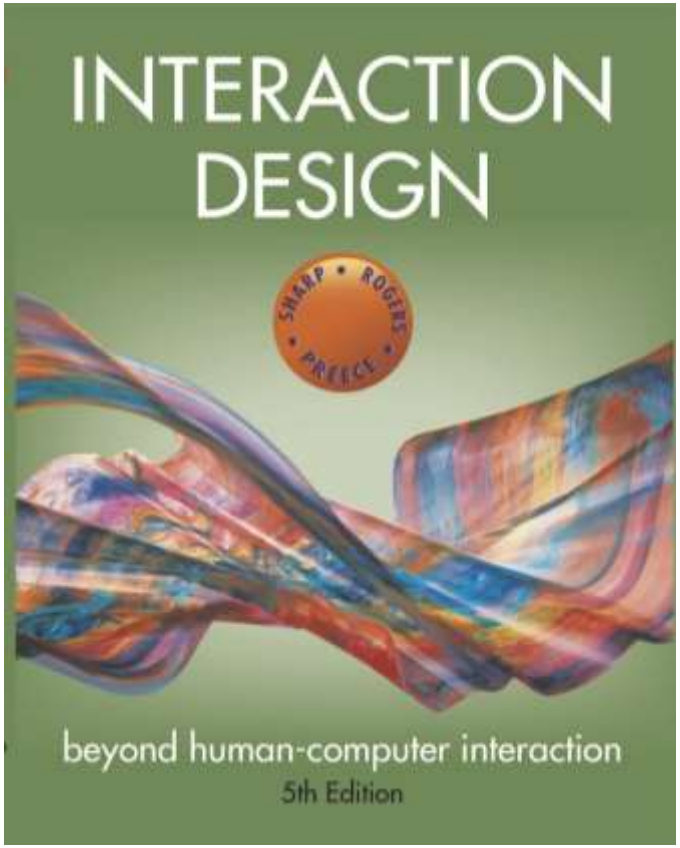
Usability lab with observers watching a user and assistant

Usability testing conditions

- Usability lab or other controlled space
- Emphasis on:
  - Selecting representative users
  - Developing representative tasks
- 5-10 users typically selected
- Tasks usually around 30 minutes
- Test conditions are the same for every participant
- Informed consent form explains procedures and deals with ethical issues

How many participants is enough for user testing?

- The number is a practical issue
- Depends on:
  - Schedule for testing
  - Availability of participants
  - Cost of running tests
- Typically 5-10 participants
- Some experts argue that testing should continue until no new insights are gained



Chapter 16
Evaluation: Inspections, Analytics, and Models

Heuristic evaluation

- Developed by Jacob Nielsen in the early 1990s
- Based on heuristics distilled from an empirical analysis of 249 usability problems
- These heuristics have been revised for current technology by Nielsen and others for:
  - Mobile devices
  - Wearables
  - Virtual worlds
  - Social media
  - …
- Design guidelines form a basis for developing heuristics

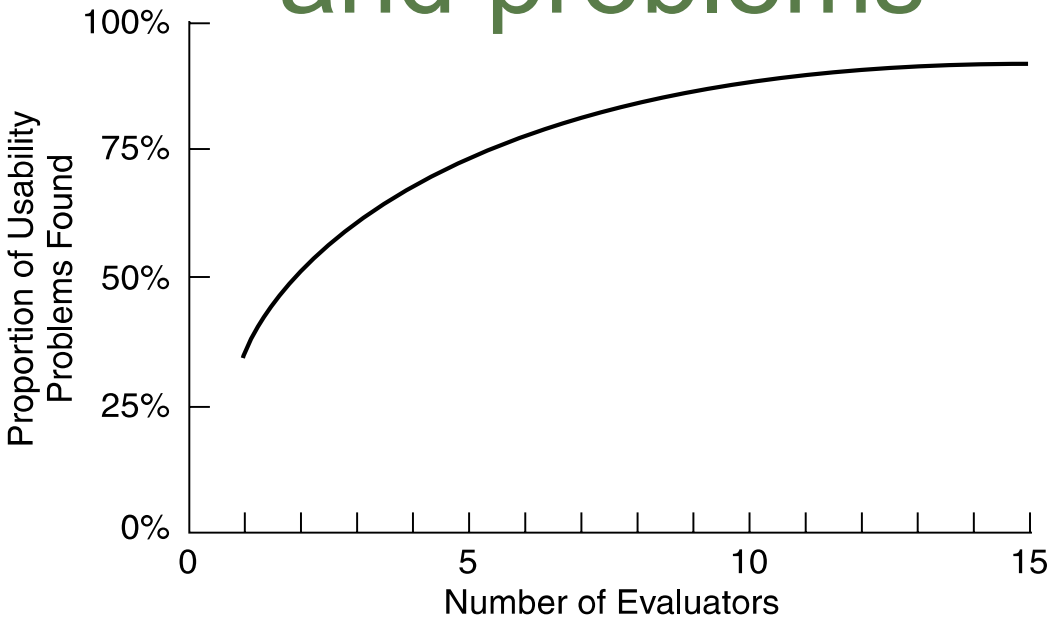
Revised version (2014) of Nielsen's original heuristics *(continued)*

- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, recover from errors
- Help and documentation

Revised version (2014) of Nielsen's original heuristics

- Visibility of system status
- Match between system and real world
- User control and freedom
- Consistency and standards
- Error prevention

Number of evaluators and problems



Curve showing the proportion of usability problems in an interface found by heuristic evaluation using different numbers of evaluators

Source: Nielsen and Mack, 1994. Courtesy of Wiley.

Number of evaluators

- Nielsen suggests that on average five evaluators identify 75-80 percent of usability problems
- Cockton and Woolrych (2001) point out that the number of users needed to find 75-80 percent of usability problems depends on the context and nature of the task problems

Heuristics for websites focus on key criteria

- Clarity
- Minimize complexity
- Provide users with context
- Provide positive and pleasurable user experience

Source: Budd, 2007

Doing heuristic evaluation

- session to tell experts what to do
- Evaluation period of 1-2 hours in which:
  - Each expert works separately
  - Take one pass to get a feel for the product
  - Take a second pass to focus on specific features
- Debriefing session in which experts work together to prioritize problems

Advantages and problems

- Few ethical and practical issues to consider because users not involved
- Can be difficult and expensive to find experts
- Best experts have knowledge of application domain and users
- Biggest problems:
  - Important problems may get missed
  - Many trivial problems are often identified, such as false alarms
  - Experts have biases

Health Informatics

Class objectives

- At the end of this class, students should be able to:
  - Identify the definitions of health informatics (HI) and the related areas of disciplines, including health information management (HIM).
  - State the key players in HI and their roles.
  - Explain educational and career opportunities in HI.

What is health informatics?

Healthcare	Informatics
The management of illness through the services of medical and allied health professionals	The sciences concerned with gathering, manipulating, storing, retrieving and classifying recorded information.

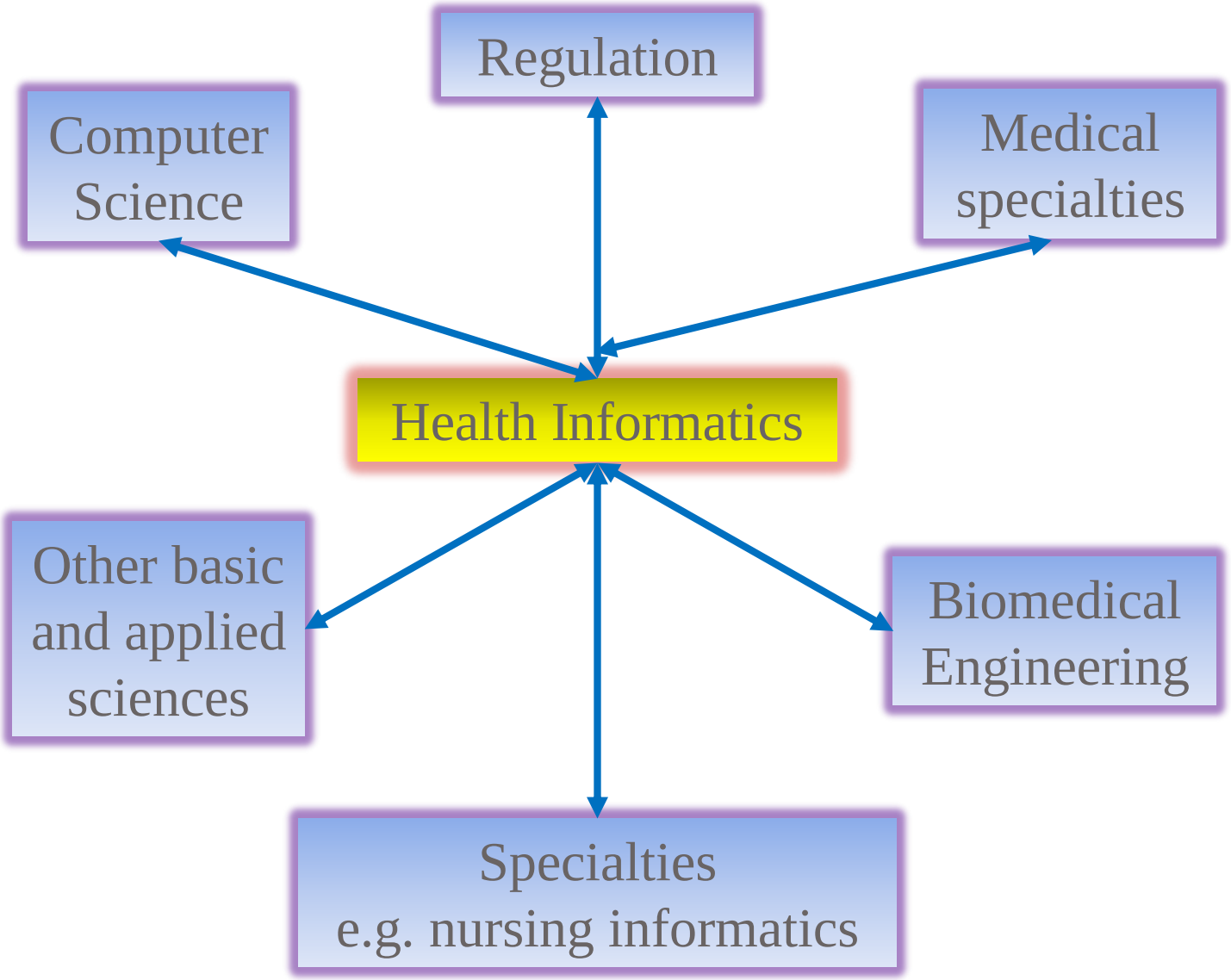
Informatics, Definition#2

'The creation, recognition, representation, collection, organization, transformation, communication, evaluation and control of information in a system…

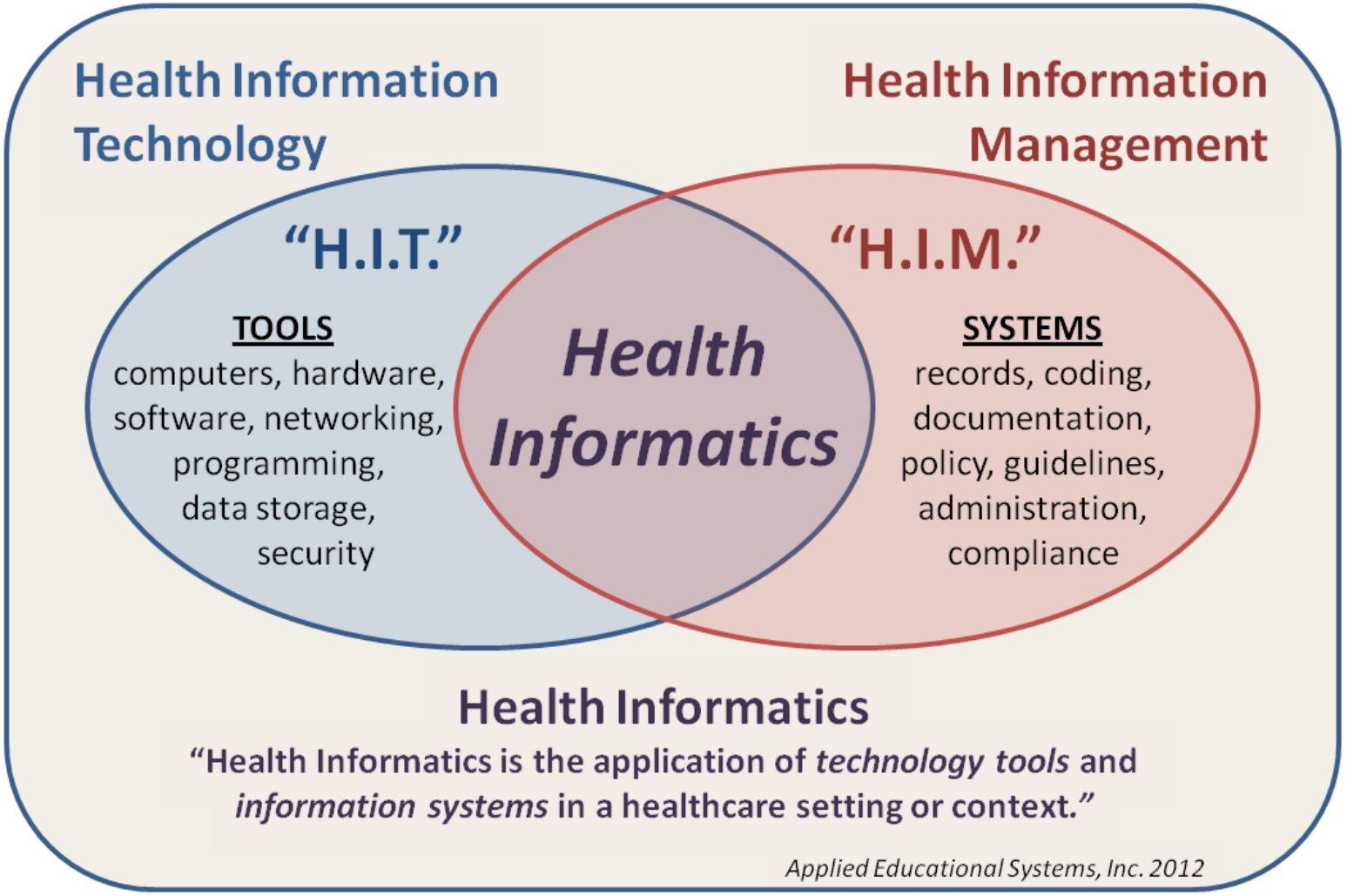
The art, science and human dimensions of information technologies.'

Indiana University School of Journalism 2002

Influences


Regulation
Computer Science
Medical specialties
Health Informatics
Other basic and applied sciences
Biomedical Engineering
Specialties
e.g. nursing informatics

HI, HIT, and HIM



Why we need health informatics?

- 11% of lab tests repeated
  - Because result is lost
- 30% of treatment orders are undocumented
- 70% of severely ill patients get right treatment
- 500% growth in number of new drugs in a decade
- Increasing patient expectation and education
- Demand for transparent processes

Who wants health informatics?

- Health professionals
  - Doctors (Primary / secondary care)
  - Nurses
  - Allied professions
- Administrators / Government
- Researchers
- Pharmaceutical Companies
- IT professionals
- And of course, patients

Health Informatics Careers

Health Information Management	Health Informatics
Medical records	Applied IT
• Medical records managers • Patient information coordinator • Billing coder • Medical biller • Privacy office/manager • Compliance officer • Data quality manager • Health information management director	• Health information system manager • Interoperability, interfacing and usability management • Applications software consulting • Database analyst/Administrators • System analyst • Application coordinators • Application support services • Knowledge manager

3.1.3 Cooperative Evaluation Technique

Monk et al. developed the co-operative evaluation technique in 1993. This approach is more than just user-centred, as it involves the user as an active participant in the evaluation process. Monk et al. advocate that it be employed as part of an iterative prototyping process and not just at the end of a product development cycle, to determine whether the design is successful or not. Rather, this approach should be adopted so that the possibilities of negative results are minimized from the start through formative evaluation.

As they state, "Co-operative evaluation is a technique to improve a user interface specification by detecting the possible usability problems in an early prototype or partial simulation. It sets down procedures by which a designer can work with the sort of people who will ultimately use the software in their daily work, so that together they can identify potential problems and their solutions".

Co-operative evaluation becomes distinctive because the collaboration occurs as users and designers evaluate the system together. Users are encouraged to ask the evaluator questions about interacting with the system and the evaluator asks them questions about their understanding of the system. This makes the procedure seems very natural to the users and requires fewer resources than more formal testing methods.

Evaluation is most useful for early feedback for redesign in a rapid iterative cycle. The aim is not to provide an exhaustive list of all the problems that could possibly be identified. Rather, it is to help designer identify the most important improvements to consider with the minimum of effort. Besides that it is cost effective in that it reveals important usability problems in a relatively short time.

For the purpose of conducting the co-operative evaluation, 6 participants were randomly selected based on their interest to participate in the evaluation process. Table 15 provides the participants details.

Table 15: Participants Details

No.	Criteria	Participant 1	Participant 2	Participant 3	Participant 4	Participant 5	Participant 6
1.	**Gender**	Male	???	???	???	???	???
2.	**Age**	30	???	???	???	???	???
3.	**Educational Level**	Master Degree	???	???	???	???	???
4.	**Programme Taken**	Ph.D Computer Science	???	???	???	???	???
5.	**Institution**	University of Jordan	???	???	???	???	???

a) Pre-Evaluation Procedures

Participants were contacted through telephone conversations asking them the possibility to participate in the co-operative evaluation. A brief introduction to the Chatsistance system was given to the participants 10 minutes before they started the evaluation, and participants were asked to read that introductory document. The document also has a list of tasks, which will be performed by the participants throughout the co-operative evaluation. Users were told that they need to think aloud when facing any problem in the system. They were also told that, each task they perform is monitored and timed.

b) Evaluation Procedures

During the evaluation session, a moderator accompanied the users to do the co-operative evaluation. A comments form shown in Appendix A was used by the moderator to write down the comments of each user for each task. Users were helped when they really face serious problems performing the tasks. The following tables show the comments form prepared by the moderator for each participant.

Table 16: Cooperative Evaluation for Chatsistance Tool for Participant No 1 **(This is just for one participant, you need to do something similar for all participants)**

Task No.	Test	Time Taken to Complete the Task	Comments
A.	**Chat Tab**		
1.	Log in to the Chatsistance tool using the following information: Server Address: 127.0.0.1 Username: Mohsen Password:123456	2 minutes and 5 seconds.	• What do you mean by "Server Address"? • The user clicked on "Connect" button 3 times then the system stopped working. • I need more information to log in. • This is not how standard chatting system work when logging in to the system. • It is not clear that users must click on "Start Server" button then on "Connect" button to log in.
2.	Participate in the current conversation by sending new message to the other users typing "This is a useful tool". Note: change "purpose of your statement" to "simple answer".	3 minutes and 14 seconds.	• From where I can change the purpose of the statement? • What is the reference and from where I can get it? • The terminologies used are not clear enough and might be misleading.
3.	Add a new topic to the Chatsistance tool and name it "Chatsistance	2 minutes and 48 seconds.	• Why do I have to write a message for me to create a new topic? • What is "Act"?

	Test".		
4.	Select any user answer then reply to it with a suitable answer.	25 seconds.	• The user completed this task without any problem.
5.	Filter the chat log to appear specific messages according to: a. Users: messages from only one user in the list (choose any user). b. Topics: messages related to a specific topic in the list (choose the topic that you have created in task 3).	2 minute.	• Why do I have to double click on each checkbox to enable or disable it?
6.	In the chat tap stop the real time update of the conversation between team members, by disabling you can freely browse through the chat log and not to be disturbed by new messages.		• The "Follow Conversation" feature is no longer available in the system although it exists in the screenshots as reported in the Chatsistance manual.
B.	**Interactions Tab**		
1.	Search the conversation in the chat log so that the output is relevant to the first topic in the topics list.	3 minutes and 20 seconds.	• The user referred to several times to the "Chat tab" thinking that he could search from there. • It is better to put the topic with the search criteria.
2.	Search the conversation in the chat log so that the output is relevant to the last user in the addressed list.	24 seconds.	• This is the same as searching for the first topic.
3.	Search the conversation in the chat log so that the output will contain the word "design" and the action is "question".	1 minute and 30 seconds.	• The user should recheck all checkboxes before start searching. • The search is confusing and not clear.
C.	**Statistics Tab**		

1.	You need to show statistics related to any question in the "Chat log", notice the results in the "Message Anchors" and "User Statistics".	2 minutes and 11 seconds.	• Is it from "Interactions tab"? • You mean I need to change the action "Act" to "Question"? • Why do I have to go to the "Interactions tab" to select criteria for getting the statistics?

It is important to compare the time taken by each participant to complete each single task compared to the default time allocated by the moderator as shown in Table 22.

Table 22: Task Completion Times in Minutes and Seconds

Task No.	Default	Participant 1	Participant 2	Participant 3	Participant 4	Participant 5	Participant 6
A. Chat Tab							
1.	00:20	02:05	03:00	01:00	01:15	01:00	03:00
2.	01:00	03:14	02:21	03:18	02:31	02:03	05:03
3.	00:30	02:48	01:41	04:20	02:00	04:15	05:45
4.	00:30	00:25	00:33	00:15	03:15	00:15	00:30
5.	01:00	02:00	02:43	02:45	02:12	04:12	06:02
B. Interactions Tab							
1.	01:00	03:20	04:00	02:00	02:00	00:15	00:20
2.	01:00	00:24	02:03	01:25	01:05	02:01	00:25
3.	00:30	01:30	01:00	03:00	02:18	04:02	04:43
C. Statistics Tab							
1.	00:30	02:11	05:22	02:03	05:00	3:00	05:32
Total Completion Time	**06:20**	**17:57**	**22:43**	**20:06**	**21:36**	**21:03**	**31:20**

c) Post-Evaluation Procedures

After completing the co-operative evaluation, participants were given a post-test questionnaire to fill in, which is shown in Appendix B. This questionnaire was important to capture their thoughts and feelings and the Chatsistance system while they were still fresh. The questionnaire was then followed by a short interview and discussion, which mainly focused on the initial modified design of the Chatsistance system. Table 23 shows the responses of the 6 participants to the post-test questionnaire.

Table 23: Participants Responses to the Post-Test Questionnaire

No.	Statement	Participant 1	Participant 2	Participant 3	Participant 4	Participant 5	Participant 6	Average
1	Chatsistance is easy to use.	4	1	1	1	3	4	**2.33**
2	Chatsistance has accomplished its goals.	5	3	3	3	3	4	**3.50**
3	Chatsistance interface is interactive.	4	3	3	1	4	4	**3.17**
4	Chat log is easy to understand and easy to follow.	4	2	2	4	4	3	**3.17**
5	It is easy to understand the functionality of the system without prior experience.	4	1	2	1	3	3	**2.33**
6	Chatsistance is enjoyable chat tool.	5	2	2	1	4	4	**3.00**
7	Interactions tap provides clear and easy to understand information.	4	3	3	2	3	4	**3.17**
8	The concept of Chatsistance was difficult to understand.	1	4	4	4	2	2	**2.83**
9	I liked the look and feel of the tool.	5	3	3	2	4	3	**3.33**
10	Chatsistance felt complete.	4	2	3	3	3	2	**2.83**
11	I felt that the tasks were difficult to complete using the tool.	1	2	3	4	3	3	**2.67**
Average		**3.73**	**2.36**	**2.64**	**2.36**	**3.27**	**3.27**	**2.94**

3.1.1 Heuristic Evaluation Technique

According to Wikipedia, heuristic evaluation is a discount usability inspection method for computer software that helps to identify usability problems in the user interface (UI) design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the "heuristics").

Heuristic evaluation is an informal usability inspection technique developed by Jakob Nielsen and his colleagues in 1994 in which experts, guided by a set of usability principles known as heuristics, evaluate whether user-interface elements, such as dialog boxes, menus, navigation structure, online help, etc., conform to the principles.

These evaluation methods are now widely taught and practiced in the New Media sector, where UIs are often designed in a short space of time on a budget that may restrict the amount of money available to provide for other types of interface testing. Nielsen's heuristic evaluation approach was used as detailed in the checklist of Usability Analysis and Design from Xerox Corporation (1995) downloaded from (http://www.stcsig.org/usability/resources/toolkit/toolkit.html#heuristics, accessed October 2010). Table 1 lists the heuristics of usability evaluation and their descriptions.

Table 1: List of Heuristics of Usability Evaluation and their Descriptions

Numbering Scheme	Heuristics	Description
H1	**Visibility of system status**	The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
H2	**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.
H3	**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.
H4	**Consistency and standards**	Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
H5	**Error prevention**	Even better than a good error message is a careful design that prevents a problem from occurring in the first place.

H6	**Recognition rather than recall**	Make 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.
H7	**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.
H8	**Aesthetic and minimalist design**	Dialogues should not contain information that 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.
H9	**Help users recognize, diagnose, and recover from errors**	Error messages should be expressed in plain language (no codes), precisely indicating the problem, and constructively suggesting a solution.
H10	**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.

In order to produce a meaningful heuristic evaluation of usability problems, 0 to 4 severity rating is associated as shown in Table 2.

Table 2: Severity Ratings and their Descriptions

Severity Rating	Description
0	I don’t agree that this is a usability problem at all.
1	Cosmetic problem only: need not be fixed unless extra time is available on project.
2	Minor usability problem: fixing this should be given low priority.
3	Major usability problem: important to fix, so should be given high priority.
4	Usability catastrophe: imperative to fix this before product can be released.

After conducting a detailed heuristic evaluation using the checklist of Usability Analysis and Design from Xerox Corporation (1995), a summary of heuristic problems are represented in Table 3.

Table 3: Summary of Heuristic Evaluation and Analysis

Data Collection and Analysis Form **Evaluator's Name:** ??? **Session Date:** ??? **Session Start Time:** ??? **Session End Time:** ???			
Location	**Heuristic Violated**	**Description of the Problem**	**Severity Rating**
Chat Tab	H1	When a user clicks on "Start Server" button and then clicks the "Connect" button the system is started. However, when the user clicks on "Start Server" button, he/she does not receive any feedback from the system with regards to the effects caused by his/her action.	4
Chat Tab	H5	When the user clicks on "Start Server" button, no feedback is received from the system. The user may click on that button several times without any response from the system, so the user do not have any guide to show them the proper sequence of actions to start chatting.	4
Chat Tab	H9	If the user clicks on "Connect" button before clicking on "Start Server" the system hangs and forces the user to close the window with no comment about the problem.	4
Chat Tab	H8	In "New Topic" part of the "Chat" tab, some fields such as "References" and "Act" are unnecessary when the user wants to add a new topic.	3
Chat Tab	H1	There is no "Recent Activities" part in this tab such as new topics added or new users. Consequently, other users will not notice the recent activities, so if a user logged into the system he/she will not have any idea about what happened when he/she is logged out.	3

Chat Tab	H4	There are two check lists in this tab. They do not work (select or deselect) by single click on any part of them, but by double click, which is something not normal and varies from the known functionality of check lists.	2
Interactions Tab	H4	The "Interaction lists" is complicated for novice users and does not support user experience.	3
Interactions Tab	H8	The shown result is quit cluttered and makes the user confused.	3
Chat, Interactions, and Statistics Tabs	H9	The system does not have an allocated part for "Help" function section that provides information about each tab of the system.	3
Statistics Tab	H1, H4, and H9	When the user clicks on “Chat log” in "Interactions" tab, suddenly the user is directed to "Statistics” page, which confuses the user.	4

Based on the data gathered in Table 3, summary of violations by heuristics is shown in Table 4, whereas summary of violations by severity rating is shown in Table 5.

Table 4: Summary of Violations by Heuristics

Heuristic Numbering Scheme	Frequency	Ratio (%)
H1	3	25.00
H2	0	0.00
H3	0	0.00
H4	3	25.00
H5	1	8.33
H6	0	0.00
H7	0	0.00
H8	2	16.67
H9	3	25.00
H10	0	0.00
Total:	**12**	**100%**

Table 5: Summary of Violations by Severity Rating

Severity Rating	Frequency	Ratio (%)
0	0	0.00
1	0	0.00
2	1	8.33
3	5	41.67
4	6	50.00
Total:	**12**	**100%**

Heuristic Evaluation - A System Checklist

Disclaimer: This list is a simplified one of the original list which was developed by Xerox corporation (© Usability Analysis & Design, Xerox Corporation, 1995) and was *downloaded from* http://www.stcsig.org/usability/topics/articles/he-checklist.html *on 28/4/2013, at 4:00pm. It has been simplified to suite the purpose it is used for, which is to evaluate the* ***Project Title*** *in order to identify current problems as experienced by the users, which is part of our graduation project that is submitted to King Abdullah II School for Information Technology, The University of Jordan. The number of questions was reduced; however, the individual questions were left intact.*

Please fill in the evaluation form below, which is a form of checklist, by writing "X" in the appropriate place which mostly describes the best answer to the corresponding criterion. This form is to be filled after you have investigated the system interface i.e. have looked at, and examined the interface. The answer to each criterion is either:

- "**0**" which means "I don't agree that this is a usability problem at all".
- "**1**" which means " Cosmetic problem only: need not be fixed unless extra time is available on project".
- "**2**" which means " Minor usability problem: fixing this should be given low priority ".
- "**3**" which means " Major usability problem: important to fix, so should be given high priority".
- "**4**" which means " Usability catastrophe: imperative to fix this before product can be released".

Thank you for your willingness to evaluate this system. Your time and effort are highly appreciated.

Group Members' Names

_____

H1. Visibility of System Status

The system should always keep user informed about what is going on, through appropriate feedback within reasonable time.

#	Review Checklist	0 1 2 3 4	Comments
1.1	Does every display begin with a title or header that describes screen contents?	( ) ( ) ( ) ( ) ( )	
1.2	Do menu instructions, prompts, and error messages appear in the same place(s) on each menu?	( ) ( ) ( ) ( ) ( )	
1.3	Is there some form of system feedback for every operator action?	( ) ( ) ( ) ( ) ( )	

1.4	Are response times appropriate to the user's cognitive processing?	( ) ( ) ( ) ( ) ( )	
1.5	Is there visual feedback in menus or dialog boxes about which choices are selectable?	( ) ( ) ( ) ( ) ( )	

H2. Match Between System and the Real World

The system should speak the user's 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.

#	Review Checklist	0 1 2 3 4	Comments
2.1	Are icons concrete and familiar?	( ) ( ) ( ) ( ) ( )	
2.2	Are menu choices ordered in the most logical way, given the user, the item names, and the task variables?	( ) ( ) ( ) ( ) ( )	
2.3	Do related and interdependent fields appear on the same screen?	( ) ( ) ( ) ( ) ( )	
2.4	When prompts imply a necessary action, are the words in the message consistent with that action?	( ) ( ) ( ) ( ) ( )	
2.5	On data entry screens, are tasks described in terminology familiar to users?	( ) ( ) ( ) ( ) ( )	

H3. User Control and Freedom

Users should be free to select and sequence tasks (when appropriate), rather than having the system do this for them. 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. Users should make their own decisions (with clear information) regarding the costs of exiting current work. The system should support undo and redo.

#	Review Checklist	0 1 2 3 4	Comments
3.1	When a user's task is complete, does the system wait for a signal from the user before processing?	( ) ( ) ( ) ( ) ( )	
3.2	Are users prompted to confirm commands that have drastic, destructive consequences?	( ) ( ) ( ) ( ) ( )	
3.3	Are character edits allowed in data entry fields?	( ) ( ) ( ) ( ) ( )	
3.4	If menu lists are long (more than seven items), can users select an item either by moving the cursor or by typing a mnemonic code?	( ) ( ) ( ) ( ) ( )	

3.5	If the system uses a pointing device, do users have the option of either clicking on menu items or using a keyboard shortcut?	( ) ( ) ( ) ( ) ( )	

H4. Consistency and Standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

#	Review Checklist	0 1 2 3 4	Comments
4.1	Has a heavy use of all uppercase letters on a screen been avoided?	( ) ( ) ( ) ( ) ( )	
4.2	Are icons labeled?	( ) ( ) ( ) ( ) ( )	
4.3	Are there no more than twelve to twenty icon types?	( ) ( ) ( ) ( ) ( )	
4.4	Does each window have a title?	( ) ( ) ( ) ( ) ( )	
4.5	Are vertical and horizontal scrolling possible in each window?	( ) ( ) ( ) ( ) ( )	
4.6	Are menu choice lists presented vertically?	( ) ( ) ( ) ( ) ( )	
4.7	Are menu titles either centered or left-justified?	( ) ( ) ( ) ( ) ( )	
4.8	Are menu items left-justified, with the item number or mnemonic preceding the name?	( ) ( ) ( ) ( ) ( )	
4.9	Do embedded field-level prompts appear to the right of the field label?	( ) ( ) ( ) ( ) ( )	
4.10	Are attention-getting techniques used with care?	( ) ( ) ( ) ( ) ( )	

H5. Help Users Recognize, Diagnose, and Recover From Errors

Error messages should be expressed in plain language (NO CODES).

#	Review Checklist	0 1 2 3 4	Comments
5.1	Is sound used to signal an error?	( ) ( ) ( ) ( ) ( )	
5.2	Are error messages worded so that the system, not the user, takes the blame?	( ) ( ) ( ) ( ) ( )	

5.3	Do error messages suggest the cause of the problem?	( ) ( ) ( ) ( ) ( )	
5.4	Do error messages indicate what action the user needs to take to correct the error?	( ) ( ) ( ) ( ) ( )	
5.5	If the system supports both novice and expert users, are multiple levels of error-message detail available?	( ) ( ) ( ) ( ) ( )	
5.6	If an error is detected in a data entry field, does the system place the cursor in that field or highlight the error?	( ) ( ) ( ) ( ) ( )	
5.7	Do error messages inform the user of the error's severity?	( ) ( ) ( ) ( ) ( )	

H6. Error Prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

#	Review Checklist	0 1 2 3 4	Comments
6.1	Are menu choices logical, distinctive, and mutually exclusive?	( ) ( ) ( ) ( ) ( )	
6.2	Are data inputs case-blind whenever possible?	( ) ( ) ( ) ( ) ( )	
6.3	Does the system prevent users from making errors whenever possible?	( ) ( ) ( ) ( ) ( )	
6.4	Does the system warn users if they are about to make a potentially serious error?	( ) ( ) ( ) ( ) ( )	
6.5	Do data entry screens and dialog boxes indicate the number of character spaces available in a field?	( ) ( ) ( ) ( ) ( )	
6.6	Do fields in data entry screens and dialog boxes contain default values when appropriate?	( ) ( ) ( ) ( ) ( )	

H7. Recognition Rather Than Recall

Make 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.

#	Review Checklist	0 1 2 3 4	Comments
7.1	For question and answer interfaces, are visual cues and white space used to distinguish questions, prompts, instructions, and user input?	( ) ( ) ( ) ( ) ( )	
7.2	Are inactive menu items grayed out or omitted?	( ) ( ) ( ) ( ) ( )	
7.3	Do data entry screens and dialog boxes indicate when fields are optional?	( ) ( ) ( ) ( ) ( )	
7.4	Are prompts, cues, and messages placed where the eye is likely to be looking on the screen?	( ) ( ) ( ) ( ) ( )	
7.5	Are field labels close to fields, but separated by at least one space?	( ) ( ) ( ) ( ) ( )	

7.6	Have items been grouped into logical zones, and have headings been used to distinguish between zones?	( ) ( ) ( ) ( ) ( )	
7.7	Are borders used to identify meaningful groups?	( ) ( ) ( ) ( ) ( )	
7.8	Is color coding consistent throughout the system?	( ) ( ) ( ) ( ) ( )	

H8. Flexibility and Minimalist Design

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. Provide alternative means of access and operation for users who differ from the "average" user (e.g., physical or cognitive ability, culture, language, etc.)

#	Review Checklist	0 1 2 3 4	Comments
8.1	If menu lists are short (seven items or fewer), can users select an item by moving the cursor?	( ) ( ) ( ) ( ) ( )	
8.2	If the system uses a pointing device, do users have the option of either clicking on fields or using a keyboard shortcut?	( ) ( ) ( ) ( ) ( )	
8.3	On data entry screens, do users have the option of either clicking directly on a field or using a keyboard shortcut?	( ) ( ) ( ) ( ) ( )	
8.4	On menus, do users have the option of either clicking directly on a menu item or using a keyboard shortcut?	( ) ( ) ( ) ( ) ( )	
8.5	In dialog boxes, do users have the option of either clicking directly on a dialog box option or using a keyboard shortcut?	( ) ( ) ( ) ( ) ( )	

H9. 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.

#	Review Checklist	0 1 2 3 4	Comments
9.1	Are all icons in a set visually and conceptually distinct?	( ) ( ) ( ) ( ) ( )	
9.2	Does each icon stand out from its background?	( ) ( ) ( ) ( ) ( )	
9.3	Does each data entry screen have a short, simple, clear, distinctive title?	( ) ( ) ( ) ( ) ( )	
9.4	Are field labels brief, familiar, and descriptive?	( ) ( ) ( ) ( ) ( )	
9.5	Are there pop-up or pull-down menus within data entry fields that have many, but well-defined, entry options?	( ) ( ) ( ) ( ) ( )	

H10. 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.

#	Review Checklist	0 1 2 3 4	Comments
10.1	Are on-line instructions visually distinct?	( ) ( ) ( ) ( ) ( )	
10.2	If menu choices are ambiguous, does the system provide additional explanatory information when an item is selected?	( ) ( ) ( ) ( ) ( )	
10.3	Is the help function visible; for example, a key labeled HELP or a special menu?	( ) ( ) ( ) ( ) ( )	
10.4	Navigation: Is information easy to find?	( ) ( ) ( ) ( ) ( )	
10.5	Presentation: Is the visual layout well designed?	( ) ( ) ( ) ( ) ( )	
10.6	Conversation: Is the information accurate, complete, and understandable?	( ) ( ) ( ) ( ) ( )	
10.7	Is the information relevant?	( ) ( ) ( ) ( ) ( )	
10.8	Can users easily switch between help and their work?	( ) ( ) ( ) ( ) ( )	
10.9	Is it easy to access and return from the help system?	( ) ( ) ( )	
10.10	Can users resume work where they left off after accessing help?	( ) ( ) ( )	

Heuristic Evaluation
A System Checklist

Primary Source
Making Computers-People Literate. © Copyright 1993.
By
Elaine Weiss
ISBN: 0-471-01877-5

System Title: **Project Title**

Secondary Source
Usability Inspection Methods. © Copyright 1994.
By
Jakob Nielsen and Robert Mack
ISBN: 1-55542-622-0

Xerox
The Document Company