First Semester 2016-2017

(Human-Computer Interaction)

IS252/CS255

M.M Marwa K.Hussein

Contents

Chapter 1: What is interaction design?

1 .I Introduction
1.2 Good and poor design
1.2.1 What to design
1.3 What is interaction design?
1.3.1the makeup of interaction design
1.3.2 Working top g of interaction design ether as a multidisciplinary team
1 3.3 Interaction design in business
1.4 What is involved in the process of interaction design?
1.5 The goals of interaction design
1.5.1 Usability goals
1.5.2 User experience goals

1.1 Introduction

In this chapter, we begin by examining what interaction design is. We look at the difference between good and poor design, highlighting how products can differ radically in their usability. We then describe what and who is involved in interaction design. In the last part of the chapter we outline core aspects of usability and how these are used to assess interactive products. An assignment is presented at the end of the chapter in which you have the opportunity to put into practice what you have read, by evaluating an interactive product using various usability criteria.

The main aims of the chapter are to:

1. Explain the difference between good and poor interaction design.
2. Describe what interaction design is and how it relates to human-computer interaction and other fields.
3. Explain what usability is.
4. Describe what is involved in the process of interaction design.
5. Outline the different forms of guidance used in interaction design.
6. Enable you to evaluate an interactive product and explain what is good and bad about it in terms of the goals and principles of interaction design.

1.2 Good and poor design

A central concern of interaction design is to develop interactive products that are sable. By this is generally meant easy to learn, effective to use, and provide an enoyable user experience. **A** good place to start thinking about how to design usable interactive products is to compare examples of well and poorly designed ones.

Through identifying the specific weaknesses and strengths of different interactive systems, we can begin to understand what it means for something to be usable or not. Here, we begin with an example of a poorly designed system -voice mail- that is used in many organizations (businesses, hotels, and universities). We then compare this with an answering machine that exemplifies good design.

1.2.1 **What to design**

Designing usable interactive products thus requires considering who is going to be using them and where they are going to be used. Another key concern is understanding the kind of ***activities*** people are doing when ***interacting*** with the products. The appropriateness of different kinds of interfaces and arrangements of input and output devices depends on what kinds of activities need to be supported. A key question for interaction design is: how do you optimize the users' inter-actions with a system, environment or product, so that they match the users' activities that are being supported and extended? One could use intuition and hope forthe best. Alternatively, one can be more principled in deciding which choices to make by basing them on an understanding of the users. This involves:

1. taking into account what people are good and bad at considering what might help people with the way they currently do things
2. thinking through what might provide quality user experiences
3. listening to what people want and getting them involved in the design using "tried and tested" user-based techniques during the design process

Activity 1.1:

How does making a phone call differ when using:
a public phone box
a cell phone?
How have these devices been designed to take into account (a) the kind of users, (b) type
of activity being supported, and (c) context of use?

1.3 What is interaction design?

By interaction design, we mean

designing interactive products to ***support people in their everyday and working lives.***

1.3.1 The makeup of interaction design

One of the biggest challenges at that time was to develop computers that could be accessible and usable by other people, besides engineers, to support tasks involving human cognition (e.g., doing sums, writing documents, managing accounts,drawing plans). To make this possible, computer scientists and psychologists became involved in designing user interfaces. Computer scientists and software engineers developed high-level programming languages (e.g., BASIC, Prolog), system architectures, software design methods, and command-based languages to help in such tasks, while psychologists provided information about human capabilities (e.g., memory, decision making).

1.3.2 Working together as a multidisciplinary team

Bringing together so many people with different backgrounds and training has meant many more ideas being generated, new methods being developed, and more creative and original designs being produced. However, the down side is the costs involved. The more people there are with different backgrounds in a design team, the more difficult it can be to communicate and progress forward the designs being generated. Why? People with different backgrounds have different perspectives and ways of seeing and talking about the world (see Figure 1.4).

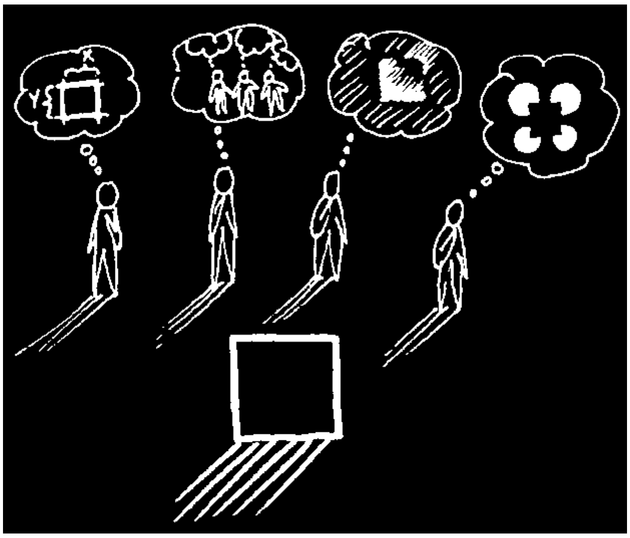


Figure 1.4 Four different team members looking at the same square, but each
Seeing it quite differently.

1.3.3 Interaction design in business

Interaction design is now a big business. In particular, website consultants, startup companies, and mobile computing industries have all realized its pivotal role in successful interactive products. To get noticed in the highly competitive field of web products requires standing out. Being able to say that your product is easy and effective to use is seen as central to this.

BOX 1.2 What's In a Name? From Interface Designers to Information Architects

Ten years ago, when a company wanted to develop an interface for an interactive product it advertised for interface designers. Such professionals were primarily involved in the design and evaluation of widgets for desktop applications. Now that the potential range of interactive products has greatly diversified, coupled with the growing realization of the importance of getting the interface right, a number of other job descriptions have begun to emerge. These include:

- interactive/interaction designers (people involved in the design of all the interactive aspects of a product, not just the graphic design of an interface)
- usability engineers (people who focus on evaluating products, using usability methods and principles)
- web designers (people who develop and create the visual design of websites, such as layouts)
- information architects (people who come up with ideas of how to plan and structure interactive products, especially websites)
- user-experience designers (people who do all the above but who may also carry out field studies to inform the design of products)

1.4 What is involved in the process of interaction design?

Essentially, the process of interaction design involves four basic activities:
1. Identifying needs and establishing requirements.
2. Developing alternative designs that meet those requirements.

3. Building interactive versions of the designs so that they can be communicated and assessed.
4. Evaluating what is being built throughout the process.
These activities are intended to inform one another and to be repeated. For example, measuring the usability of what has been built in terms of whether it is easy to use provides feedback that certain changes must be made or that certain requirements have not yet been met.

In addition to the four basic activities of design, there are three key characteristics of the interaction design process:
1. Users should be involved through the development of the project.
2. Specific usability and user experience goals should be identified, clearly documented, and agreed upon at the beginning of the project.
3. Iteration through the four activities is inevitable.

1.5 The goals of interaction design

Part of the process of understanding users' needs, with respect to designing an interactive system to support them, is to be clear about your primary objective. Is it to design a very efficient system that will allow users to be highly productive in their work, or is it to design a system that will be challenging and motivating so that it supports effective learning, or is it something else? We call these top level concerns usability goals and user experience goals. The two differ in terms of how they are operationalized,i.e., how they can be met and through what means. Usability goals are concerned with meeting specific usability criteria (e.g., efficiency) and user experience goals are largely concerned with explicating the quality of the user experience (e.g., to be aesthetically pleasing).

1.5.1 Usability goals

To recap, usability is generally regarded as ensuring that interactive products are easy to learn, effective to use, and enjoyable from the user's perspective. It involves optimizing the interactions people have with interactive products to enable them to carry out their activities at work, school, and in their everyday life. More specifically, usability is broken down into the following goals:

- effective to use (effectiveness)
- efficient to use (efficiency)
- safe to use (safety)
- have good utility (utility)
- easy to learn (learnability)
- easy to remember how to use (memorability)

For each goal, we describe it in more detail

Effectiveness is a very general goal and refers to how good a system is at doing what it is supposed to do.

Efficiency refers to the way a system supports users in carrying out their tasks.

Safety involves protecting the user from dangerous conditions and undesirable situations. In relation to the first ergonomic aspect, it refers to the external conditions where people work

Utility refers to the extent to which the system provides the right kind of functionality so that users can do what they need or want to do.

Learnability refers to how easy a system is to learn to use. It is well known that people don't like spending a long time learning how to use a system. They want to get started straight away and become competent at carrying out tasks without too much effort. This is especially so for interactive products intended for everyday use (e.g., interactive TV, email) and those used only infrequently (e.g., video conferencing).

Memorability refers to how easy a system is to remember how to use, once learned. This is especially important for interactive systems that are used infrequently. If users haven't used a system or an operation for a few months or longer,they should be able to remember or at least rapidly be reminded how to use it.

1.5.2 User experience goals

The realization that new technologies are offering increasing opportunities for sup- porting people in their everyday lives has led researchers and practitioners to con-sider further goals. The emergence of technologies (e.g., virtual reality, the web, mobile computing) in a diversity of application areas (e.g., entertainment, educa- tion, home, public areas) has brought about a much wider set of concerns. As well as focusing primarily on improving efficiency and productivity at work, interaction design is increasingly concerning itself with creating systems that are:

- satisfying
- enjoyable
- fun
- entertaining
- helpful
- motivating
- aesthetically pleasing
- supportive of creativity
- rewarding
- emotionally fulfilling

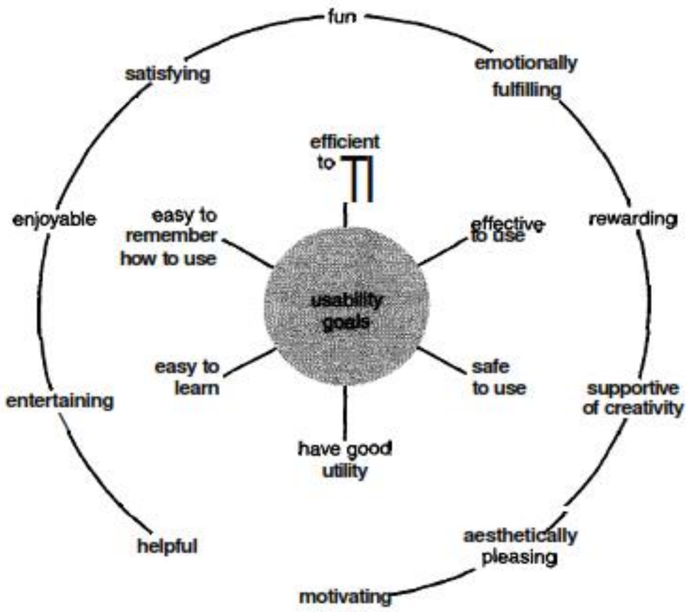


Figure 1.7 Usability and user experience goals. Usability goals are central to interaction design and are operationalized through specific criteria. User experience goals are shown in the outer circle and are less clearly defined.

Summary

In this chapter we have looked at what interaction design is and how it has evolved. We examined briefly its makeup and the various processes involved. We pointed out how the notion of usability is fundamental to interaction design. This was explained in some detail,describing what it is and how it is operationalized to assess the appropriateness, effectiveness, and quality of interactive products. **A** number of high-level design principles were also introduced that provide different forms of guidance for interaction design.

Key points

- Interaction design is concerned with designing interactive products to support people in reading their everyday and working lives.
- Interaction design is multidisciplinary, involving many inputs from wide-reaching disciplines and fields.
- Interaction design is now big business: many companies want it but don't know how to do it.
- Optimizing the interaction between users and interactive products requires taking into account a number of interdependent factors, including context of use, type of task, and kind of user.
- Interactive products need to be designed to match usability goals like ease of use and learning.
- User experience goals are concerned with creating systems that enhance the user experience in terms of making it enjoyable, fun, helpful, motivating, and pleasurable.
- Design and usability principles, like feedback and simplicity, are useful heuristics for analyzing and evaluating aspects of an interactive product.

Chapter 2: Understanding users

2.1 Introduction

In this chapter we examine some of the core cognitive aspects of interaction design. specifically, we consider what humans are good and bad at and show how this knowledge can be used to ***inform*** the design of technologies that both ***extend*** human capabilities and***compensate*** for their weaknesses. We also look at some of the influential cognitively based conceptual frameworks that have been developed for explaining the way humans interact with computers. (Other ways of conceptualizing human behavior that focus on the social and affective aspects of interaction design are presented in the following two chapters.)

The main aims of this chapter are to:

- Explain what cognition is and why it is important for interaction design.
- Describe the main ways cognition has been applied to interaction design.
- Provide a number of examples in which cognitive research has led to the design of more effective interactive products.
- Explain what mental models are.
- Give examples of conceptual frameworks that are useful for interaction design.
- Enable you to try to elicit a mental model and be able to understand what it means.

2.2 What is cognition?

Cognition is what goes on in our heads when we carry out our everyday activities.

It involves cognitive processes, like thinking, remembering, learning, daydreaming, decision making, seeing, reading, writing and talking. As Figure 3.1 indicates, there are many different kinds of cognition. Norman (1993) distinguishes between two general modes: experiential and reflective cognition. The former is a state of mind in which we perceive, act, and react to events around us effectively and effortlessly. It requires reaching a certain level of expertise and engagement. Examples include driving a car, reading a book, having a conversation, and playing a video game. In contrast, reflective cognition involves thinking, comparing, and decision-making. This kind of cognition is what leads to new ideas and creativity. Examples include designing, learning, and writing a book. Norman points out that both modes are essential for everyday life but that each requires different kinds of technological support.

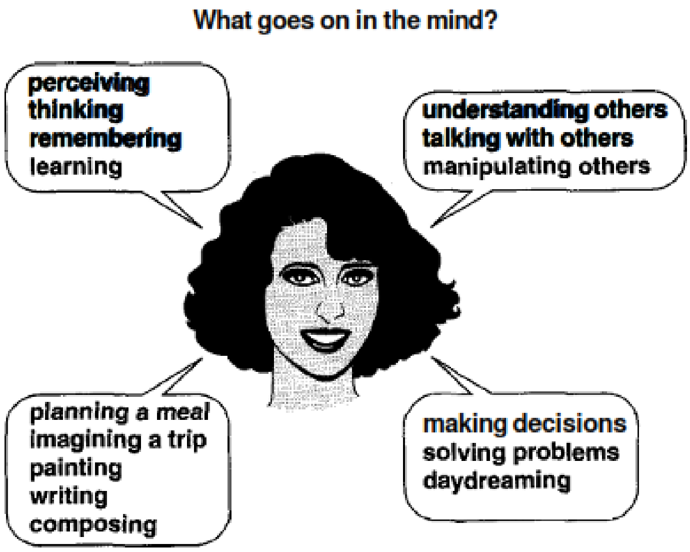


Figure 3.1 What goes on in the mind?

Cognition has also been described in terms of specific kinds of processes. These include:

- attention
- perception and recognition
- memory
- learning
- reading, speaking, and listening
- problem solving, planning, reasoning, decision making

❖ ***Attention*** is the process of selecting things to concentrate on, at a point in time,from the range of possibilities available. Attention involves our auditory andlor visual senses. An example of auditory attention is waiting in the dentist's waiting room for our name to be called out to know when it is our time to go in. An example of attention involving the visual senses is scanning the football results in a newspaper to attend to information about how our team has done. Attention allows us focus on information that is relevant to what we are doing. The extent to which this process is easy or difficult depends on (i) whether we have clear goals and (ii)whether the information we need is salient in the environment:

(i)***Our goals*** If we know exactly what we want to find out, we try to match this with the information that is available.

(ii)***Information presentation*** The way information is displayed can also greatly influence how easy or difficult it is to attend to appropriate pieces of information.

❖ ***Perception*** refers to how information is acquired from the environment, via the different sense organs (e.g., eyes, ears, fingers) and transformed into experiences of objects, events, sounds, and tastes (Roth, 1986). It is a complex process, involving other cognitive processes such as memory, attention, and language. Vision is the most dominant sense for sighted individuals, followed by hearing and touch. With respect to interaction design, it is important to present information in a way that can be readily perceived in the manner intended. For example, there are many ways to design icons. The key is to make them easily distinguishable from one another and to make it simple to recognize what they are intended to represent .

- ***Memory*** involves recalling various kinds of knowledge that allow us to act appropriately. It is very versatile, enabling us to do many things. For example, it allows us to recognize someone's face, remember someone's name, recall when we last met them and know what we said to them last. Simply, without memory we would not be able to function.
- ***Learning*** can be considered in terms of (i) how to use a computer-based application or (ii) using a computer-based application to understand a given topic. Jack Carroll (1990) and his colleagues have written extensively about how to design interfaces to help learners develop computer-based skills.

A main observation is that people find it very hard to learn by following sets of instructions in a manual. Instead, they much prefer to "learn through doing." GUIs and direct manipulation interfaces are good environments for supporting this kind of learning by supporting exploratory interaction and importantly allowing users to "undo" their actions, i.e., return to previous state if they make a mistake by clicking on the wrong option. Carroll has also suggested that another way of helping learners is by using a "training-wheelsapproach. This involves restricting the possible functions that can be carried out by a novice to the basics and then extending these as the novice becomes more experienced. The underlying rationale is to make initial learning more tractable, helping the learner focus on simple operations before moving on to more complex ones.

- ***Reading, speaking and listening:*** these three forms of language processing have both similar and different properties. One similarity is that the meaning of sentences or phrases is the same regardless of the mode in which it is conveyed. For example, the sentence "Computers are a wonderful invention" essentially has the same meaning whether one reads it, speaks it, or hears it. However, the ease with which people can read, listen, or speak differs depending on the person, task, and context. For example, many people find listening much easier than reading. Specific differences between the three modes include:

- Written language is permanent while listening is transient. It is possible to reread information if not understood the first time round. This is not possible with spoken information that is being broadcast.
- Reading can be quicker than speaking or listening, as written text can be rapidly scanned in ways not possible when listening to serially presented spoken words.
- Listening requires less cognitive effort than reading or speaking. Children, especially, often prefer to listen to narratives provided in multimedia or web-based learning material than to read the equivalent text online.
- Written language tends to be grammatical while spoken language is often ungrammatical. For example, people often start a sentence and stop in mid-sentence, letting someone else start speaking.
- There are marked differences between people in their ability to use language. Some people prefer reading to listening, while others prefer listening. Likewise, some people prefer speaking to writing and vice versa.
- Dyslexics have difficulties understanding and recognizing written words, making it hard for them to write grammatical sentences and spell correctly.
- People who are hard of hearing or hard of seeing are also restricted in the way they can process language.

❖ ***Problem-solving, planning, reasoning and decision-making*** are all cognitive processes involving reflective cognition. They include thinking about what to do, what the options are, and what the consequences might be of carrying out a given action. They often involve conscious processes (being aware of what one is thinking about), discussion with others (or oneself), and the use of various kinds of artifacts, (e.g., maps, books, and pen and paper).

Comparing different sources of information is also common practice when seeking information on the web. For example, just as people will phone around for a range of quotes, so too, will they use different search engines to find sites that give the best deal or best information. If people have knowledge of the pros and cons of different search engines, they may also select different ones for different kinds of queries. For example, a student may use a more academically oriented one when looking for information for writing an essay, and a more commercially based one when trying to find out what's happening in town.

2.3 Applying knowledge from the physical world to the digital world

A well known approach to applying knowledge about everyday psychology to interaction design is to emulate, in the digital world, the strategies and methods people commonly use in the physical world. An assumption is that if these work well in the physical world, why shouldn't they also work well in the digital world? In certain situations, this approach seems like a good idea. Examples of applications that have been built following this approach include electronic post -it notes in the form of "stickies," electronic "to-do" lists,and email reminders of meetings and other events about to take place. The stickies application displays different colored notes on the desktop in which text can be inserted, deleted, annotated, and shufffed around, enabling people to use them to remind themselves of what they need to do-analogous to the kinds of externalizing they do when using paper stickies.

Emulating real-world activity at the interface can be a powerful design strategy, provided that new functionality is incorporated that extends or supports the users in their tasks in ways not possible in the physical world. The key is really to understand the nature of the problem being addressed in the electronic world in relation to the various coping and externalizing strategies people have developed to deal with the physical world.

2.4 Conceptual frameworks for cognition

In this section we examine three of people's coping strategies in the physical world to the digital world., which each have a different perspective on cognition:

- mental models
- information processing
- external cognition

2.4.1 Mental models

What happens when people are learning and using a system is that they develop knowledge of how to use the system and, to a lesser extent, how the system works. These two kinds of knowledge are often referred to as a user's mental model.

Having developed a mental model of an interactive product, it is assumed that people will use it to make inferences about how to carry out tasks when using the interactive product. Mental models are also used to fathom what to do when something unexpected happens with a system and when encountering unfamiliar systems. The more someone learns about a system and how it functions, the more their mental model develops.For example, TV engineers have a "deep" mental model of how TVs work that allows them to work out how to fix them.

2.4.2 information processing

Another approach to conceptualizing how the mind works has been to use metaphors and analogies.**A** number of comparisons have been made, including conceptualizing the mind as a reservoir, a telephone network, and a digital computer. One prevalent metaphor from cognitive psychology is the idea that the mind is an information processor. Information is thought to enter and exit the mind through a series of ordered processing stages (see Figure 3.11). Within these stages, various processes are assumed to act upon mental representations. Processes include comparing and matching. Mental representations are assumed to comprise images, mental models, rules, and other forms of knowledge.

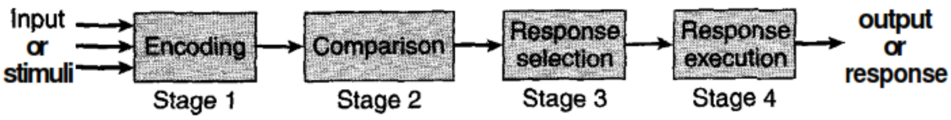


Figure 3.11 Human information processing model.

Several researchers have argued that existing information processing approaches are too impoverished:

The traditional approach to the study of cognition is to look at the pure intellect, isolated from distractions and from artificial aids. Experiments are performed in closed, isolated rooms, with a minimum of distracting lights or sounds, no other people to assist with the task, and no aids to memory or thought. The tasks are arbitrary ones, invented by the researcher. Model builders build simulations and descriptions of these isolated situations.

The theoretical analyses are self -contained little structures, isolated from the world, isolated from any other knowledge or abilities ofthe person. (Norman, 1990, p. 5)

Instead, there has been an increasing trend to study cognitive activities in the Context in which they occur, analyzing cognition as it happens "in the wild" (Hutchins, 1995). **A** central goal has been to look at how structures in the environment can both aid human cognition and reduce cognitive load. A number of alternative frameworks have been proposed, including external cognition and distributed cognition.

2.4.3 External cognition

People interact with or create information through using a variety of external representations, e.g., books, multimedia, newspapers, web pages, maps, diagrams, notes, drawings, and so on. Furthermore, an impressive range of tools has been developed throughout history to aid cognition, including pens, calculators, and computer-based technologies. The ***combination*** of external representations and physical tools have greatly extended and supported people's ability to carry out cognitive activities (Norman, 1993). Indeed, they are such an integral part that it is difficult to imagine how we would go about much of our everyday life without them.

External cognition is concerned with explaining the cognitive processes involved When we interact with different external representations (Scaife and Rogers, 1996). **A** main goal is to explicate the cognitive benefits of using different representations for different cognitive activities and the processes involved. The main ones include:

1. externalizing to reduce memory load
2. computational offloading
3. annotating and cognitive tracing

1 .Externalizing to reduce memory load

A number of strategies have been developed for transforming knowledge into external representations to reduce memory load. One such strategy is externalizing things we find difficult to remember, such as birthdays, appointments, and addresses. Diaries, personal reminders and calendars are examples of cognitive artifacts that are commonly used for this purpose, acting as external reminders of what we need to do at a given time (e.g., buy a card for a relative's birthday).

Externalizing, therefore, can help reduce people's memory burden by:

- reminding them to do something (e.g., to get something for their mother's birthday)
- reminding them of what to do (e.g., to buy a card)
- reminding them of when to do something (send it by a certain date)

2. Computational offloading

Computational offloading occurs when we use a tool or device in conjunction with an external representation to help us carry out a computation. An example is using pen and paper to solve a math problem.

3. Annotating and cognitive tracing

Another way in which we externalize our cognition is by modifying representations to reflect changes that are taking place that we wish to mark. For example, people often cross things off in a to-do list to show that they have been completed. They may also reorder objects in the environment, say by creating different piles as the nature of the work to be done changes. These two kinds of modification are called annotating and cognitive tracing:

Annotating involves modifying external representations, such as crossing off or underlining items.

2.5 Informing design: from theory to practice

Theories, models, and conceptual frameworks provide abstractions for thinking about phenomena. In particular, they enable generalizations to be made about cognition across different situations. For example, the concept of mental models provides a means of explaining why and how people interact with interactive products in the way they do across a range of situations. The information processing model has been used to predict the usability of a range of different interfaces.

Theory in its pure form, however, can be difficult to digest. The arcane terminology and jargon used can be quite off-putting to those not familiar with it. It also requires much time to become familiar with it-something that designers and engineers can't afford when working to meet deadlines.

Researchers have tried to help out by making theory more accessible and practical. This has included translating it into:

- design principles and concepts

- design rules
- analytic methods
- design and evaluation methods

Summary

This chapter has explained the importance of understanding users, especially their cognitive aspects. It has described relevant findings and theories about how people carry out their everyday activities and how to learn from these when designing interactive products. It has provided illustrations of what happens when you design systems with the user in mind and what happens when you don't.

It has also presented a number of conceptual frameworks that allow ideas about cognition to be generalized across different situations.

Key points

- Cognition comprises many processes, including thinking, attention, learning, memory, perception, decision-making, planning, reading, speaking, and listening.
- The way an interface is designed can greatly affect how well people can perceive, attend,learn, and remember how to carry out their tasks.
- The main benefits of conceptual frameworks and cognitive theories are that they can explain user interaction and predict user performance.
- The conceptual framework of mental models provides a way of conceptualizing the user's understanding of the system.
- Research findings and theories from cognitive psychology need to be carefully reinterpreted in the context of interaction design to avoid oversimplification and misapplication.

Chapter 3 :The process of interaction design

3.1. Introduction

In this chapter, we raise and answer these kinds of questions and discus[s the four basic activities and key characteristics of the interaction design process[that were introduced in Chapter 1. We also introduce a lifecycle model of interaction design that captures these activities and characteristics.

The main aims of this chapter are to:

1. Consider what 'doing' interaction design involves.
2. Ask and provide answers for some important questions about the interaction design process.
3. Introduce the idea of a lifecycle model to represent a set of activities and how they are related.
4. Describe some lifecycle models from software engineering and HCI and discuss how they relate to the process of interaction design.
5. Present a lifecycle model of interaction design.

3.2 What is interaction design about?

Interaction design involves developing a plan which is informed by the product's intended use, target domain, and relevant practical considerations. Alternative designs need to be generated, captured, and evaluated by users. For the evaluation to be successful, the design must be expressed in a form suitable for users to interact with.

3.2.1Four basic activities of interaction design

Four basic activities for interaction design were introduced in Chapter 1. These are: identifying needs and establishing requirements, developing alternative designs that meet those requirements, building interactive versions so that they can be communicated and assessed, and evaluating them, i.e., measuring their acceptability. They are fairly generic activities and can be found in other designs disciplines too.

We will be expanding on each of the basic activities of interaction design in the next two chapters. Here we give only a brief introduction to each.

Identifying needs and establishing requirements

In order to design something to support people, we must know who our target users are and what kind of support an interactive product could usefully provide. These needs form the basis of the product's requirements and underpin subsequent design and development. This activity is fundamental to a user centered approach, and is very important in interaction design.

Developing alternative designs

This is the core activity of designing: actually suggesting ideas for meeting the requirements. This activity can be broken up into two sub-activities: conceptual design and physical design. Conceptual design involves producing the conceptual model for the product, and a conceptual model describes what the product should do, behave and look like. Physical design considers the detail of the product including the colors, sounds, and images to use, menu design, and icon design. Alternatives are considered at every point.

Building interactive versions of the designs

Interaction design involves designing interactive products. The most sensible way for users to evaluate such designs, then, is to interact with them. This requires an interactive version of the designs to be built, but that does not mean that a software version is required. There are different techniques for achieving "interaction," not all of which require a working piece of software. For example, paper-based prototypes are very quick and cheap to build and are very effective for identifying problems in the early stages of design, and through role-playing users can get a real sense of what it will be like to interact with the product.

Evaluating designs

Evaluation is the process of determining the usability and acceptability of the product or design that is measured in terms of a variety of criteria including the number of
errors users make using it, how appealing it is, how well it matches the requirements, and so on. Interaction design requires a high level of user involvement throughout development, and this enhances the chances of an acceptable product being delivered. In most design situations you will find a number of activities concerned with quality assurance and testing to make sure that the final product is "fit-for-purpose." Evaluation does not replace these activities, but complements and enhances them. The activities of developing alternative designs, building interactive versions of the design, and evaluation are intertwined: alternatives are evaluated through the interactive versions of the designs and the results are feedback into further design. This iteration is one of the key characteristics of the interaction design process.

3.2.2 Three key characteristics of the interaction design process

There are three characteristics that we believe should form a key part of the interaction design process. These are: a user focus, specific usability criteria, and iteration. The need to
focus on users has been emphasized throughout this book, so you will not be surprised to see that it forms a central plank of our view on the interaction design process.

Specific usability and user experience goals should be identified, clearly documented, and agreed upon at the beginning of the project. They help designers to choose between different alternative designs and to check on progress as the product is developed.

Iteration allows designs to be refined based on feedback. As users and designers engage with the domain and start to discuss requirements, needs, hopes and aspirations, then different insights into what is needed, what will help, and what is feasible will emerge.

3.3 Lifecycle models: showing how the activities are related

Understanding what activities are involved in interaction design is the first step to being able to do it, but it is also important to consider how the activities are related to one another so that the full development process can be seen. The term *lifecycle model* is used to represent a model that captures a set of activities and how they are related. Sophisticated models also incorporate a description of when and how to move from one activity to the next and a description of the deliverables for each activity. The reason such models are popular is that they allow developers, and particularly managers, to get an overall view of the development effort so that progress can be tracked, deliverables specified, resources allocated, targets set, and so on.

3.3.1 A simple lifecycle model for interaction design

We see the activities of interaction design as being related as shown in Figure 6.7. This model incorporates iteration and encourages a user focus. While the outputs from each activity are not specified in the model. Most projects start with identifying needs and requirements. The project may have arisen because of some evaluation that has been done, but the lifecycle of the new (or modified) product can be thought of as starting at this point. From this activity, some alternative designs are generated in an attempt to meet the needs and requirements that have been identified. Then interactive versions of the designs are developed and evaluated. Based on the feedback from the evaluations, the team may need to return to identifying needs or refining requirements, or it may go straight into redesigning. It may be that more than one alternative design follows this iterative cycle in parallel with others, or it may be that one alternative at a time is considered. Implicit in this cycle is that the final product will emerge in an evolutionary fashion from a rough initial idea through to the finished product.

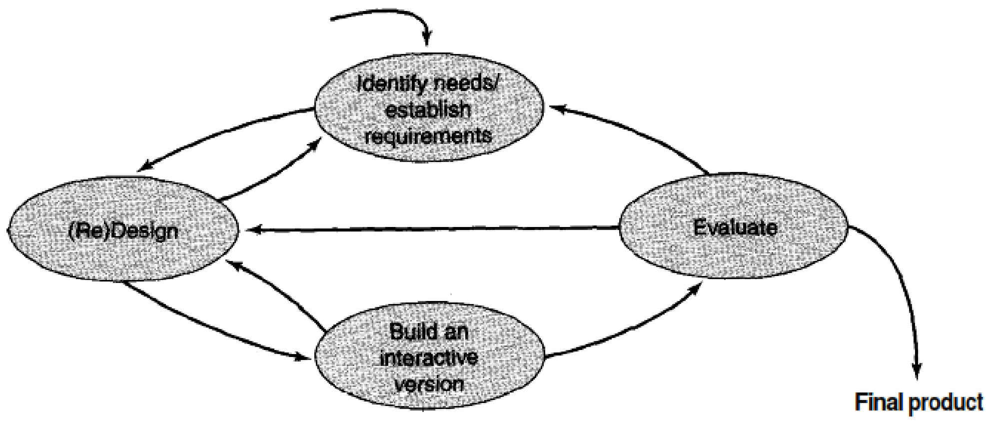


Figure 6.7 A simple interaction design model.

3.3.2 Lifecycle models in software engineering

Software engineering has spawned many lifecycle models, including the waterfall, the spiral, and rapid applications development (RAD).

The waterfall lifecycle model

The waterfall lifecycle was the first model generally known in software engineering and forms the basis of many lifecycles in use today. This is basically a linear model in which each step must be completed before the next step can be started (see Figure 6.8)

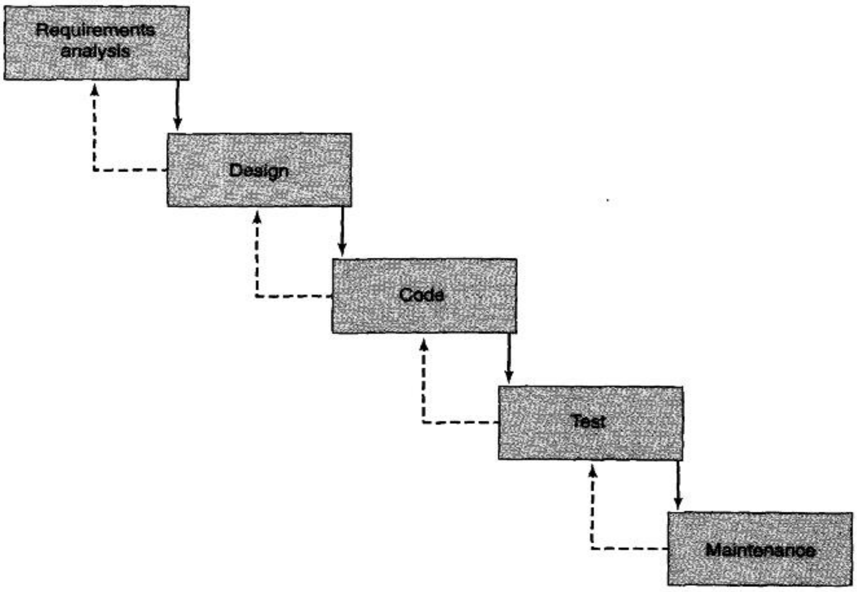
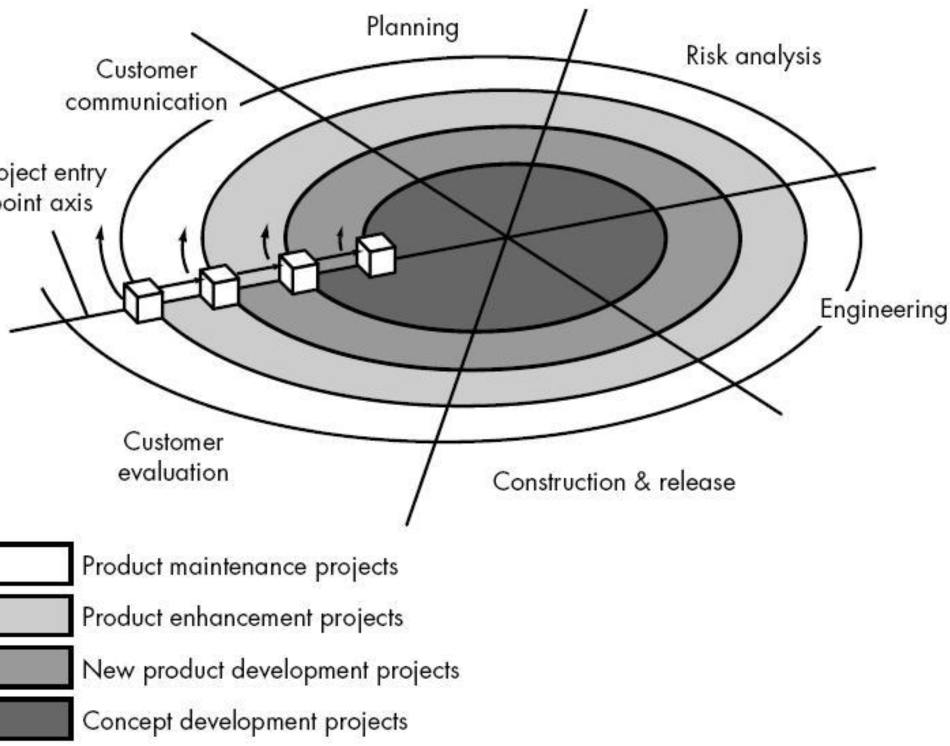


Figure 6.8 The waterfall lifecycle model of software development.

The spiral lifecycle model

For many years, the waterfall formed the basis of most software developments, but in 1988 Barry Boehm (1988) suggested the spiral model of software development (see Figure 6.9). Two features of the spiral model are immediately clear from Figure 6.9: risk analysis and prototyping. The spiral model incorporates them in an iterative framework that allows ideas and progress to be repeatedly checked and evaluated. Each iteration around the spiral may be based on a different lifecycle model and may have different activities.

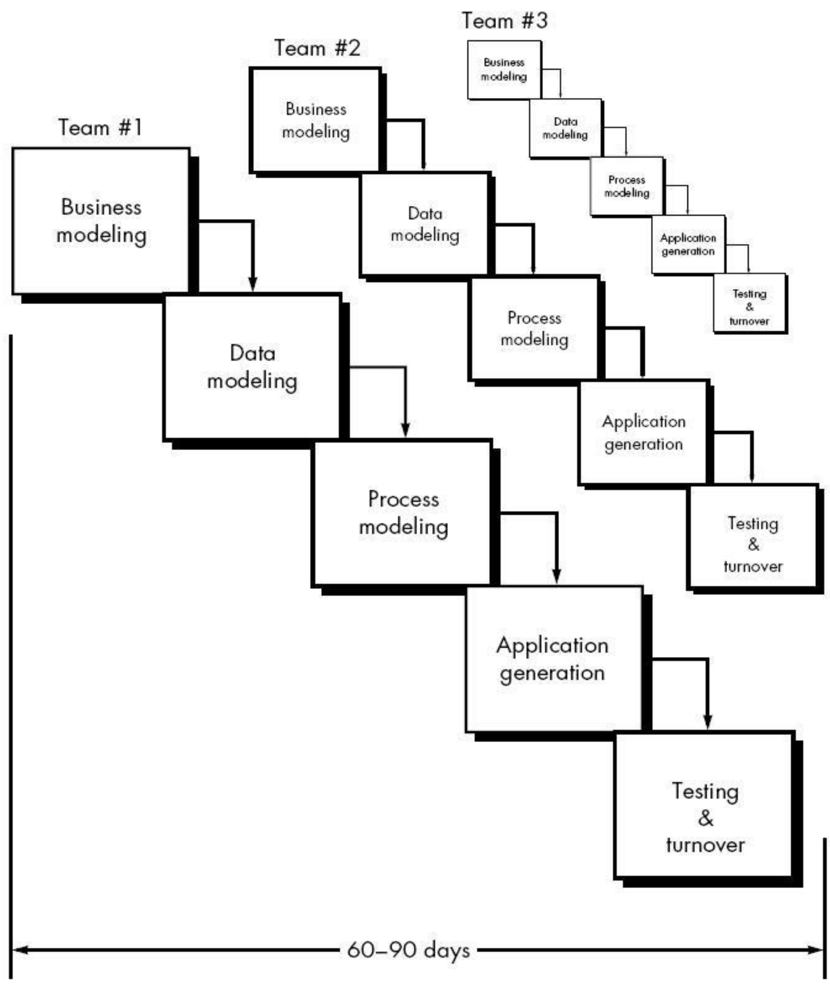
FIGURE
A typical spiral model



(RAD) approach attempts to take a user-centered view and to minimize the risk caused by requirements changing during the course of the project. The ideas behind RAD began to emerge in the early 1990s, also in response to the inappropriate nature of the linear lifecycle models based on the waterfall. Two key features of A RAD project are:

- Time-limited cycles of approximately six months, at the end of which a system or partial system must be delivered. This is called time-boxing. In effect, this breaks down a large project into many smaller projects that can deliver products incrementally, and enhances flexibility in terms of the development techniques used and the maintainability of the final system.
- JAD (Joint Application Development) workshops in which users and developers come together to thrash out the requirements of the system (Wood and Silver, 1995). These are intensive requirements-gathering sessions which difficult issues are faced and decisions are made. Representatives each identified stakeholder group should be involved in each workshop that all the relevant views can be heard.

FIGURE
The RAD model



3.3.3 Lifecycle models in HCI

Another of the traditions from which interaction design has emerged is the field of HCI (human -computer interaction). Fewer lifecycle models have arisen from this field than from software engineering and, as you would expect, they have a stronger tradition of user focus. We describe two of these here. The first one, the Star, was derived from empirical work on understanding how designers tackled HCI design problems. This represents a very flexible process with evaluation at its core. In contrast, the second one, the usability engineering lifecycle, shows a more structured approach and hails from the usability engineering tradition.

The *Star* Lifecycle Model

In 1989, the Star lifecycle model was proposed by Hartson and Hix (1989) (see Figure 6.13). This emerged from some empirical work they did looking at how interface designers went about their work. They identified two different modes of activity: analytic mode and synthetic mode. The former is characterized by such notions as top -down, organizing, judicial,and formal, working from the systems view towards the user's view; the latter is characterized by such notions as bottom-up, free-thinking, creative and ad *hoc,* working from the user's view towards the systems view. Interface designers move from one mode to another when designing a similar behavior has been observed in software designers (Guindon,1990).

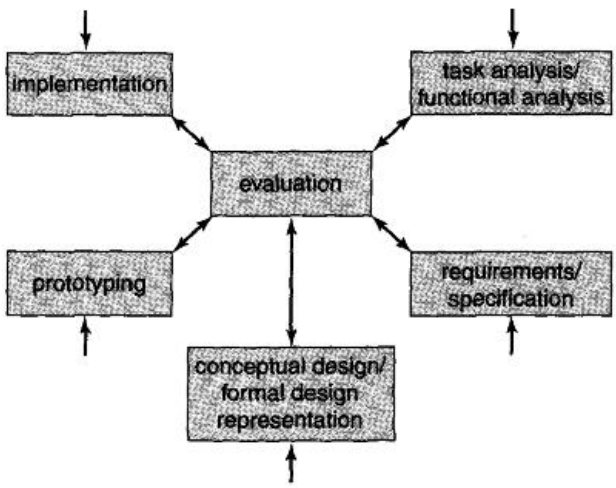


Figure 6.13 The Star lifecycle model.

The Usability Engineering Lifecycle

The Usability Engineering Lifecycle was proposed by Deborah Mayhew in 1999 (Mayhew, 1999). The lifecycle itself has essentially three tasks: requirements analysis, design, testing, development, and installation, with the middle stage being the largest and involving many subtasks (see Figure 6.14). Note the production of a set of **usability** goals in the first task. Mayhew suggests that these goals be captured in a style guide that is then used throughout the project to help ensure that the usability goals are adhered to.

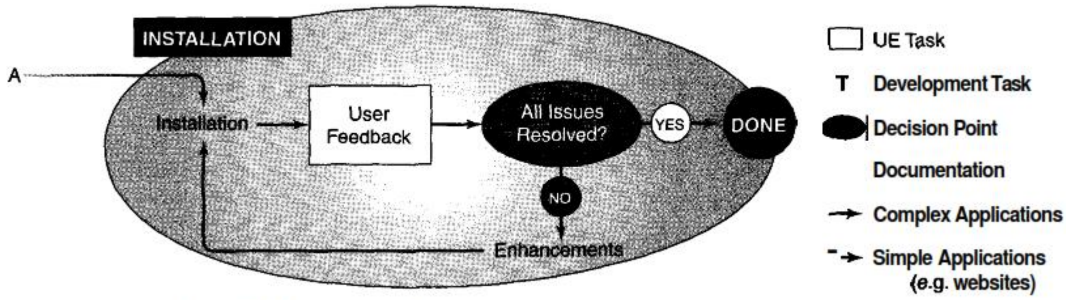


Figure 6.14 *(continued).*

Summary

In this chapter, we have looked at the process of interaction design, i.e., what activities are required in order to design an interactive product, and how lifecycle models show te relationships between these activities.

A simple interaction design model consisting of four activities was introduced and issues surrounding the identification of users, generating alternative designs, and evaluating designs were discussed. Some lifecycle models from software engineering and **HCI** were introduced.

Key points

- The interaction design process consists of four basic activities: identifying needs and establishing requirements, developing alternative designs that meet those requirements, building interactive versions of the designs so that they can be communicated and assessed, and evaluating them.
- Key characteristics of the interaction design process are explicit incorporation of user involvement, iteration, and specific usability criteria.
- Before you can begin to establish requirements, you must understand who the users are and what their goals are in using the device.
- Looking at others' designs provides useful inspiration and encourages designers to consider alternative design solutions, which is key to effective design.
- Usability criteria, technical feasibility, and users' feedback on prototypes can all be used to choose among alternatives.
- Prototyping is a useful technique for facilitating user feedback on designs at all stages.
- Lifecycle models show how development activities relate to one another.
- The interaction design process is complementary to lifecycle models from other fields.

Chapter 4 : Introducing evaluation

4.1 Introduction

This chapter begins by discussing ***what*** evaluation is, ***why*** evaluation is important, and ***when*** to use different evaluation techniques and approaches. Then a case study is presented about the evaluation techniques used by Microsoft researchers and the Fred Hutchinson Cancer Research Center in developing HutchWorld (Cheng et al., 2000), a virtual world to support cancer patients, their families, and friends. This case study is chosen because it illustrates how a range of techniques is used during the development of a new product. It introduces some of the practical problems that evaluators encounter and shows how iterative product development is informed by a series of evaluation studies. The HutchWorld study also lays the foundation for the evaluation framework. The main aims of this chapter are to:

- Explain the key concepts and terms used to discuss evaluation.
- Discuss and critique the HutchWorld case study.
- Examine how different techniques are used at different stages in the development of HutchWorld.
- Show how developers cope with real-world constraints in the development of HutchWorld.

4.2 What, why, and when to evaluate

Users want systems that are easy to learn and to use as well as effective, efficient, safe, and satisfying. Being entertaining, attractive, and challenging, etc. is also essential for some products. ***So,*** knowing what to evaluate, why it is important, and when to evaluate are key skills for interaction designers.

4.2.1 What to evaluate

There is a huge variety of interactive products with a vast array of features that need to be evaluated. Some features, such as the sequence of links to be followed to find an item on a website, are often best evaluated in a laboratory, since such a setting allows the evaluators to control what they want to investigate. Other aspects, such as whether a collaborative toy is robust and whether children enjoy interacting with it, are better evaluated in natural settings, so that evaluators can see what children do when left to their own devices.

4.2.2 Why you need to evaluate

Just as designers shouldn't assume that everyone is like them, they also shouldn't presume that following design guidelines guarantees good usability, Evaluation is

needed to check that users can use the product and like it.
Tognazzini points out that there are five good reasons for investing in user testing:
1. Problems are fixed before the product is shipped, not after.
2. The team can concentrate on real problems, not imaginary ones.
3. Engineers code instead of debating.
4. Time to market is sharply reduced.
5. Finally, upon first release, your sales department has a rock-solid design it can sell without having to pepper their pitches with how it will all actually work in release 1.1 or 2.0.

4.2.3 When to evaluate

The product being developed may be a brand-new product or an upgrade of an existing product. If the product is new, then considerable time is usually invested in market research. Designers often support this process by developing mockups of the potential product that are used to elicit reactions from potential users. As well as helping to assess market need, this activity contributes to understanding users' needs and early requirements uation is to assess how well a design fulfills users' needs and whether users like it.

In the case of an upgrade, there is limited scope for change and attention is focused on improving the overall product. This type of design is well suited to usability engineering in which evaluations compare user performance and attitudes with those for previous versions. Some products, such as office systems, go through many versions, and successful products may reach double digit version numbers. In contrast, new products do not have previous versions and there may be nothing comparable on the market, so more radical changes are possible if evaluation results indicate a problem.

Evaluations done during design to check that the product continues to meet users' needs are know as *formative evaluations.* Evaluations that are done to assess the success of a finished product, such as those to satisfy a sponsoring agency or to check that a standard is being upheld, are know as *summative evaluation.* Agencies such as National Institute of Standards and Technology (NIST) in the USA, the In ternational Standards Organization (ISO) and the British Standards Institute (BSI) set standards by which products produced by others are evaluated.

4.3 HutchWorld case study

HutchWorld is a distributed virtual community developed through collaboration between Microsoft's Virtual Worlds Research Group and librarians and clinicians at the Fred Hutchinson Cancer Research Center in Seattle, Washington. The system enables cancer patients, their caregivers, family, and friends to chat with one another, tell their stories, discuss their experiences and coping strategies, and gain emotional and practical support from one another (Cheng et. al.,2000). The design team decided to focus on this particular population because caregivers and cancer patients are socially isolated: cancer patients must often avoid physical contact with others because their treatments suppress their immune systems. Similarly, their caregivers have to be careful not to transmit infections to patients.

4.3.1How the design team got started: early design ideas

Before developing this product, the team needed to learn about the patient experience at the Fred Hutchinson Center. For instance, what is the typical treatment process, what resources are available to the patient community, and what are the

needs of the different user groups within this community? They had to be particularly careful about doing this because many patients were very sick. Cancer patients also typically go through bouts of low emotional and physical energy.
Caregivers also may have difficult emotional times, including depression, exhaustion, and stress. Furthermore, users vary along other dimensions, such as education and experience with computers, age and gender and they come from different cultural backgrounds with different expectations.
The development team decided that HutchWorld should be available for patients any time of day or night, regardless of their geographical location.The team's informal visits to the Fred Hutchinson Center led to the development of an early prototype. They followed a user-centered development methodology. Having got a good feel for the users' needs, the team brainstormed different ideas for an organizing theme to shape the conceptual design a conceptual model possibly based on a metaphor. After much discussion, they decided to make the design resemble the outpatient clinic lobby of the Fred Hutchinson Cancer ResearchCenter. By using this real-world metaphor, they hoped that the users would easily infer what functionality was available in HutchWorld from their knowledge of the real clinic. The next step was to decide upon the kind of communication environment to use. Should it be synchronous or asynchronous? Which would support social and affective communications best? **A** synchronous chat environment was selected because the team thought that this would be more realistic and personal than an asynchronous environment. They also decided to include 3D photographic avatars so that users could enjoy having an identifiable online presence and could easily recognize each other.
The prototype was reviewed with users throughout early development and was later tested more rigorously in the real environment of the Hutch Center using a variety of techniques.
A Microsoft product called V-Chat was used to develop second interactive prototype with the subset of the features in the preliminary design ,however, only the lobby was fully developed.
Before testing could begin, the team had to solve some logistical issues. There were two key questions. Who would provide training for the testers and help for the patients? And how many systems were needed for testing and where should they be placed? As in many high -tech companies, the Microsoft team was used to short, market-driven production schedules, but this time they were in for a shock.
Organizing the testing took much longer than they anticipated, but they soon learned to set realistic expectations that were in synch with hospital activity and the unexpected delays that occur when working with people who are unwell.

4.3.2 How was the testing done?

The team ran two main sets of user tests. The first set of tests was informally run onsite at the Fred Hutchinson Center in the hospital setting. After observing the system in use on computers located in the hospital setting, the team redesigned the software and then ran formal usability tests in the usability labs at Microsoft.

Test 1 : **Early observations onsite**

In the informal test at the hospital, six computers were set up and maintained by Hutch staff members. A simple, scaled -back prototype of HutchWorld was built using the existing product, Microsoft V-Chat and was installed on the computers, which patients and their families from various hospital locations used. Over the course of several months, the team trained Hutch volunteers and hosted events in the V-Chat prototype. The team observed the usage of the space during unscheduled times, and they also observed the general usage of the prototype.

Test 1 : **What was learned?**

This V-Chat test brought up major usability issues. First, the user community was relatively small, and there were never enough participants in the chat room for successful communication-a concept known as ***critical mass.*** In addition, many of the patients were not interested in or simultaneously available for chatting. Instead, they preferred asynchronous communication, which does not require an immediate response. Patients and their families used the computers for email, journals, discussion lists, and the bulletin boards largely because they could be used at any time and did not require others to be present at the same time. The team learned that a strong asynchronous base was essential for communication.

The team also observed that the users used the computers to play games and to search the web for cancer sites approved by Hutch clinicians. This information was not included in the virtual environment, and so users were forced to use many different applications. **A** more "unified" place to find all of the Hutch content was desired that let users rapidly swap among a variety of communication, information, and entertainment tasks.

Test 1 : **The redesign**

Based on this trial, the team redesigned the software to support more asynchronous communication and to include a variety of communication, information, and entertainment areas. They did this by making HutchWorld function as a portal that provides access to information -retrieval tools, communication tools, games, and other types of entertainment. Other features were incorporated too, including email, a bulletin board, a text-chat, a web page creation tool, and a way of checking to see if anyone is around to chat with in the 3D world.

Test 2: Usability tests

After redesigning the software, the team then ran usability tests in the Microsoft usability labs. Seven participants (four male and three female) were tested. Four of these participants had used chat rooms before and three were regular users. All had browsed the web and some used other communications software. The participants were told that they would use a program called HutchWorld that was designed to provide support for patients and their families. They were then given five minutes to explore HutchWorld. They worked independently and while they explored they provided a running commentary on what they were looking at, what they were thinking, and what they found confusing. This commentary was recorded on video and so were the screens that they visited, so that the Microsoft evaluator, who watched through a one -way mirror, had a record of what happened for later analysis. Participants and the evaluator interacted via a microphone and speakers. When the five-minute exploration period ended, the participants were asked to complete a series of *structured tasks* that were designed to test particular features of the HutchWorld interface.

These tasks focused on how participants dealt with their virtual identity; that is, how they represented themselves and were perceived by others communicated with others got the information they wanted found entertainment

4.3.3 Was it tested again?

Following the usability testing, there were more rounds of observation and testing with six new participants, two males and four females. These tests followed the same general format as those just described but this time they tested multiple users at once, to ensure that the virtual world supported multiuser interactions. The tests were also more detailed and focused. This time the results were more positive, but of course there were still usability problems to be fixed. Then the question arose: what to do next? In particular, had they done enough testing (see Dilemma)?

After making a few more fixes, the team stopped usability testing with specific tasks. But the story didn't end here. The next step was to show HutchWorld to cancer patients and caregivers in a focus-group setting at the Fred Hutchinson Cancer Research Center to get their feedback on the final version. Once the team made adjustments to HutchWorld in response to the focus-group feedback, the final step was to see how well HutchWorld worked in a real clinical environment. It was therefore taken to a residential building used for long term patient and family stays that was fully wired for Internet access. Here, the team observed what happened when it was used in this natural setting. In particular, they wanted to find out how HutchWorld would integrate with other aspects of patients' lives, particularly with their medical care routines and their access to social support. This informal observation allowed them to examine patterns of use and to see who used which parts of the system, when, and why.

4.3.4 Looking *to* the future

Future studies were planned to evaluate the effects of the computers and the software in the Fred Hutchinson Center. The focus of these studies will be the social support and wellbeing of patients and their caregivers in two different conditions.There will be a control condition in which users (i.e., patients) live in the residential building without computers and an experimental condition in which users live in similar conditions but with computers, Internet access, and HutchWorld. The team will evaluate the user data (performance and observation) and surveys collected in the study to investigate key questions, including:

- How does the computer and software impact the social wellbeing of patients and their caregivers?
- What type of computer-based communication best supports this patient community?
- What are the general usage patterns? i.e., which features were used and at
- what time of day were they used, etc.?

How might any medical facility use computers and software like Hutch-World to provide social support for its patients and caregivers?

4.4 Discussion

In both HutchWorld and the 1984 Olympic Messaging System, a variety of evaluation techniques were used at different stages of design to answer different questions.

"Quick and dirty" observation, in which the evaluators informally examine how a prototype is used in the natural environment, was very useful in early design. Following this with rounds of usability testing and redesign revealed important usability problems. However, usability testing alone is not sufficient.

Field studies were needed to see how users used the system in their natural environments, and sometimes the results were surprising. For example, in the OMS system users from different cultures behaved differently. A key issue in the HutchWorld study was how use of the system would fit with patients' medical routines and changes in their physical and emotional states. Users' opinions also offered valuable insights. After all, if users don't like a system, it doesn't matter how successful the usability testing is: they probably won't use it. Questionnaires and interviews were used to collect user's opinions.

An interesting point concerns not only how the different techniques can be used to address different issues at different stages of design, but also how these techniques complement each other. Together they provide a broad picture of the system's usability and reveal different perspectives. In addition, some techniques are better than others for getting around practical problems. This is a large part of being a successful evaluator. In the HutchWorld study, for example, there were not many users, so the evaluators needed to involve them sparingly. For example, a technique requiring 20 users to be available at the same time was not feasible in the HutchWorld study, whereas there was no problem with such an approach in the OMS study. Furthermore, the OMS study illustrated how many different techniques, some of which were highly opportunistic, can be brought into play depending on circumstances. Some practical issues that evaluators routinely have to address include:

- what to do when there are not many users
- how to observe users in their natural location (i.e., field studies) without disturbing them
- having appropriate equipment available
- dealing with short schedules and low budgets
- not disturbing users or causing them duress or doing anything unethical
- collecting "useful" data and being able to analyze it
- selecting techniques that match the evaluators' expertise

Summary

The aim of this chapter was to introduce basic evaluation concepts that will be revisited and built on in the next four chapters. We selected the HutchWorld case study because it illustrates how a team of designers evaluated a novel system and coped with a variety of practical constraints. It also shows how different techniques are needed for different purposes and how techniques are used together to gain different perspectives on a product's usability. Thisstudy highlights how the development team paid careful attention to usability and user experience goals as they designed and evaluated their system.

Key points

- Evaluation and design are very closely integrated in user-centered design.
- Some of the same techniques are used in evaluation as in the activity of establishing requirements and identifying users' needs, but they are used differently (e.g., interviews and questionnaires, etc.).
- Triangulation involves using combinations of techniques in concert to get different perspectives or to examine data in different ways.

  Dealing with constraints, such as gaining access to users or accommodating users' routines, is an important skill for evaluators to develop.

عوامل التفاعل بين الانسان والحاسوب

ان العوامل الرئيسية التي يجب ان يؤخذ ينظر الاعتبار عند تصميم التفاعل هي:

1- عوامل بيئية : (الضجيج – الحرارة – الاضاءة – التهوية)
2- عوامل تنظيمية : (التدريب وتصميم الوظيفة والسياسات والقوانين وتنظيم العمل)
3- عوامل الراحة : (طريقة الجلوس ووضع الجهاز)
4- معالجات ادراكية ومقدرات المستخدم : (الدوافع – الاستمتاع – الرضا – الشخصية – مستوى الخبرة)
5- عوامل الصحة والامان : (الضغط – التنفس- الصداع – المعوقات العضلية)
6- عوامل المهام : مستوى سهولة المهام او تعقيدها وحداثتها وتخصيص المهام وتكرارها والمهارات اللازمة لادائها ومكوناتها
7- واجهة المستخدم: اجهزة الدخل والخرج وهيكل الحوار واستخدام الالوان والايقونات والاوامر والرسومات واللغات الطبيعية والصور الثلاثية الابعاد ومواد دعم المستخدم والوسائط المتعددة
8- القيودات: (التكلفة- الزمن- الميزانية – العمالة – المعدات- المكان)
9- وظائفية النظام: (الاليات – البرمجيات – التطبيقات)
10- عوامل انتاجية: تقليل متطلبات العمالة وتقليل زمن الانتاج وتطوير الافكار الابتكارية التي تؤدي الى منتج جديد.

ان بعض العوامل ذات علاقة مباشرة بالمستخدم مثل الراحة والصحة وبعضها ذو علاقة بعمل المستخدم مثل بيئة العمل والبعض الآخر ذو علاقة بالتقنية المستخدمة . ومما يجعل التحليل اكثر تعقيدا هو ان كثير من العوامل تتفاعل مع بعضها البعض بصورة ثابتة.

أسس تصميم التفاعل هي:

1- الرؤية Visibility : هي العلاقة المحسوسة بين التحكمات وآثارها.
2- التحملية Affordance: هي من خصائص الكيانات التي يستخدمها البشر فلكل كيان او آلة تحملية معينة, وتعرف التحملية بأنها الوظائف والمعالجات التي يتحملها النظام او يمكن عملها على كيان معين.
مثال:\
المرئية : انظمة التحكم داخل السيارة.
التحملية: للباب قابلية تحمل, والكرسي كذلك.

اهمية التفاعل بين الانسان والحاسوب:

1- زيادة الإنتاجية: Productivity
من العناصر المهمة في التفاعل بين الإنسان والحاسوب هي الفوائد المالية الملموسة التي تجدها المؤسسات من استخدامها للنظام وهي ما يسمى بزيادة الإنتاجية

2- تحاشي المخاطر:
هذه المخاطر هي كل ما يتسبب في أضرار للمستخدم وقد يكون من بين المخاطر انهيار النظام نفسه

3- واجهة المستخدم الرسومية :(Graphical User Interface GUI)

تؤمن للمستخدم التفاعل مع الحاسب باستخدام أغراض وصور رسومية غالباً ما تتكون من عناصر تحكم ونوافذ وقوائم منبثقة إضافة لنصوص توجه المستخدم لاستخدام أحداث مخصصة مثل نقر الفأرة إضافة لإدخال نصوص ليقوم الحاسب بما يريد المستخدم. جميع الأفعال والمهام التي يمكن للحاسب تنفيذها تتم عن طريق التطبيق المباشر لأحداث على العناصر الرسومية (عناصر التحكم).

مبادئ تصميم واجهة الاستخدام

- مبدأ الهيكلة:

التصميم يجب أن ينظم واجهة الاستخدام بشكل هادف، بطرق مفيدة يجب أن يستند إلى الوضوح من الأكثر أهمية إلى الأقل أهمية

- مبدأ البساطة :

التصميم يجب أن يكون بسيط، لا يعني أن يكون عادي جداً ولكن أن تكون المهام فيه سهلة والتواصل ما بين المستخدم والنظام يكون بشكل واضح وبسيط من ناحية لغة المستخدم التي يتعامل فيها مع النظام وتزويده بالطرق المختصرة والتي لها علاقة بالمهام والإجراءات الطويلة بحيث يصل إليها بشكل سريع.

مثال (اختصارات لوحة التحكم)

- مبدأ الوضوح:

التصميم يجب أن يجعل جميع الاختيارات والمواد المطلوبة للمهمة واضحة بدون صرف انتباه المستخدم بالمعلومات الزائدة أو الغريبة. كذلك وضوح المهام التي يعمل بها المستخدم عن المهام التي لا يعمل بها حالياً .التصاميم الجيدة لا تغرق المستخدم بالبدائل او بالتشويش عليه عن طريق تزويده بالمزيد من المعلومات الغيرمطلوبة منه.

- مبدأ التجاوب:

التصميم يجب أن يبقي المستخدمين مطلعين على الأحداث الجارية للمهام التي يقومون بها في الموقع والتفسيرات التي تنتج عن هذه الأحداث، كرسائل الأخطاء التي لها علاقة بالمهمة أو اللغة جزئية المهمة التي لم تمرر بشكل سليم، يجب أن تكون المستخدمة واضحة وبسيطة ومألوفة لدى المستخدم.

- مبدا التحمل:

التصميم يجب أن يكون مرن ومتساهل بعدة زوايا، بخفض نسبة الأخطاء، متقبل للاستعمال السيء/الخاطئ من قبل المستخدم كالأخطاء أو التوقفات التي يمكن أن تظهر مقابل إعادة مهمة مطلوبة للمستخدم أو عدم إكمالها بحيث يمنع التطبيق إظهار أخطاء أو عمليات غير مفهومة إلى المستخدم في مقابل إظهار رسائل الخطأ والتنبيات المتعلقة بالأشياء التي يقوم بها المستخدم في الموقع أو التطبيق.

المعرفة هي الإدراك والوعي وفهم الحقائق أو اكتساب المعلومة عن طريق التجربة أو من خلال التأمل في طبيعة الأشياء وتأمل النفس أو من خلال الإطلاع على تجارب الآخرين وقراءة استنتاجاتهم، المعرفة مرتبطة بالبديهة والبحث لاكتشاف المجهول وتطوير الذات وتطوير التقنيات.

المعرفة يحددها قاموس أوكسفورد الإنكليزي بأنها : (أ) الخبرات والمهارات المكتسبة من قبل شخص من خلال التجربة أو التعليم ؛ الفهم النظري أو العملي لموضوع، (ب) مجموع ما هو معروف في مجال معين ؛ الحقائق والمعلومات، الوعي أو الخبرة التي اكتسبتها من الواقع أو من القراءة أو المناقشة, (ج) المناقشات الفلسفية في بداية التاريخ مع أفلاطون صياغة المعرفة بأنها "الإيمان الحقيقي المبرر". بيد أنه لا يوجد تعريف متفق عليه واحد من المعارف في الوقت الحاضر، ولا أي احتمال واحد، وأنه لا تزال هناك العديد من النظريات المتنافسة.

كما تعرف **المعرفة** أيضا بأنها:

وصف لحالة أو عملية لبعض الجوانب الحياتية بالنسبة لأشخاص أو مجموعات مستعدة لها، فمثلا إذا كنت "أعرف" أنها ستمطر، فإنني سوف آخذ مظلتي معي عند الخروج.

والمعرفة أيضاً هي ثمرة التقابل والاتصال بين الذات المدركة وموضوع مدرك، وتتميز من باقي معطيات الشعور، من حيث أنها تقوم في آن واحد على التقابل والاتحاد الوثيق بين هذين الطرفين.[1] وقد قدم لنا الأستاذ الدكتور عبد الوهاب المسيري تعريفاً اجرائياً لكلمة معرفة وهو أقرب إلى الأذهان لدارس الفلسفة بقوله "المعرفي هو الكلي والنهائي وتعبير الكلية هنا يفيد الشمول والعموم في حين أن النهائية للوجود تعنى غائيته وأخره وأقصى مايمكن أن يبلغه الشئ ويمكن التوصل للبعد المعرفي لأي خطاب أو أي ظاهرة من خلال دراسة ثلاثة عناصر أساسية: