Tutorial Letter 102/0/2026

Human–Computer Interaction

INF1520

Year Module

Department Information Systems

IMPORTANT INFORMATION

Please register on myUnisa, activate your myLife e-mail account and make sure that you have regular access to the myUnisa module website, INF1520-2026-Y, as well as your group website.

Note: This is a fully online module. It is, therefore, only available on myUnisa.

BARCODE

CONTENTS

1. Lesson: Module Orientation

1.1 Welcome Message

Welcome to

INF1520

Human-Computer Interaction

Welcome to this module on human-computer interaction (HCI). We hope that you will find it interesting and stimulating. INF1520 Human-Computer Interaction is a module that will broaden your horizons to understand why pages are designed in the way that they are and how they should be designed.

This module focuses on enhancing the quality of the interaction between humans and machines and to systematically apply our knowledge about human purposes, capabilities and limitations as well as machine capabilities and limitations.

1.2 Overview

The module INF1520 is about Human-Computer Interaction (HCI). The study of HCI is done to determine how we can make computer technology more usable for people. The overall purpose of the module is to:

- enhance the quality of the interaction between human and machine by systematically applying our knowledge about the human purpose, capabilities and limitations as well as about machine capabilities and limitations.
- develop or improve productivity as well as the functionality, safety, utility, effectiveness, efficiency and usability of systems that include computers (Preece et al 2007; Preece et al 2014)

This requires an understanding of the:

- computer technology involved.
- people who interact with the computer technology
- design of interactive systems and interfaces which are usable.
- broader impact of computer technology on society and on our social, personal and working environments

These four strands form the focus of this module.

Outcomes

On completing this module, you should have knowledge and an understanding of:

- the history of human-computer interaction and its current status
- the differences between users and designers
- the perceptual, cognitive and physical characteristics of computer users

- factors such as culture, personality and age that may influence the design and usability of software especially in organisations
- typical mistakes designers make and guidelines and principles to address these
- the different types of interfaces that computer systems can have
- techniques to evaluate interactive systems
- the impact of computers on society

1.3 The way forward

INF1520 is an online module, so you need to use myUnisa to study and complete the learning activities for this module. Because this is an online module, you have to go online to see your study materials and read about what needs to be done for the module. Go to the website https://my.unisa.ac.za and log in with your student number and password. You will see INF1520-26-Y in the row of modules in the orange blocks across the top of the webpage. Remember to also check in the **More** tab if you cannot find it in the orange blocks. Click on the module you want to open. You need to visit the websites on myUnisa for INF1520 frequently. The website for your module is INF1520-26-Y.

This TUT102 serves as your study guide and learning material.

There is no prescribed textbook for the module.

Lecturer(s)

You may contact lecturers by mail, e-mail or telephone. We recommend the use of e-mail. The COSALLF tutorial letter contains the names and contact details of your INF1520 lecturer. You may also make an appointment to see a lecturer, but this has to be done well in advance. Please mention your student number in all communication with your lecturers.

Department

In the meantime, if you would like to speak to a lecturer, you may contact the secretary of the School of Computing at 011 471 2816. Remember to mention your student number. This is for academic queries only. Please do not contact the School about missing tutorial matter, cancellation of a module, payments, enquiries about the registration of assignments, and so on. For all queries not related to the content of the module, you have to contact the relevant department as indicated in the brochure *Study @ Unisa*.

University

For more information on myUnisa, consult the brochure *Study @ Unisa*, which you received with your study material: www.unisa.ac.za/brochures/studies. This brochure contains information about computer laboratories, the library, myUnisa, assistance with study skills, et cetera. It also contains the contact details of several Unisa departments, for example, Examinations, Assignments, Despatch, Finances and Student Administration. Remember to mention your student number when contacting the university.

1.4 Assessment

The assignment details are included in the Tutorial Letter 101, which will be posted online and will be available under the study material tool on the myUnisa page.

1.5 Study Plan

The suggested study plan is given in Tutorial 101. Please feel free to adapt this to your own circumstances.

1.6 Relevant material to study

The module makes use of a study guide which is captured in this TUT102. We have **not** prescribed a textbook for this module. Each lesson will provide an indication of the content to be studied. Each assignment also provides the relevant lesson from the TUT102 to be used in order to complete the assignment. Prof CJ van Staden updated this TUT102 in 2019.

1.7 Additional resources

There are no recommended books for this module.

7. LESSONS 1 – 5

2.1 Lesson 1: Introduction to Human-Computer Interaction

The content of this lesson is as follows:

CONTENTS

LESSON 1 – OUTCOMES

After studying this lesson tool, you should:

- understand the historical context of HCI:
  - Where did HCI innovations and philosophy come from?
  - Who were the major people involved?
  - What were the important systems?
  - How did the ideas move from the laboratory to the market?
- be able to define what HCI is
- be able to define important concepts related to HCI
- understand the current context of HCI
- be able to describe future directions in the field
- understand why HCI is considered to be a multidisciplinary subject and list the related disciplines such as business applications, mobile applications and games

1.1 Introduction

Computers and computer software are created for people to use. They should therefore be designed in a way that allows the intended user to use them successfully for the intended purpose and with the least amount of effort. To design a successful system, the designers must know how to support the tasks that the user will perform with such system. They must understand why users need the system, what tasks they will want to perform with the system, what knowledge they might have (or lack) that may influence their interaction with the system, and how the system fits into the user's existing context.

The term *human-computer interaction* (HCI) was adopted in the mid-1980s to denote a new field of study concerned with studying and improving the effectiveness and efficiency of computer use. Today it is a multidisciplinary subject with computer science, psychology and cognitive science at its core (Dix, Finlay, Abowd & Beal 2004). When HCI became one of the domains of cognitive science research in the 1970s, the idea was to apply cognitive science methods to software development (Carroll 2003). General principles of perception, motor activity, problem-solving, language and communication were viewed as sources that could guide design. Although HCI has now expanded into a much broader field of study, it is still true that knowledge of cognitive psychology can help designers to understand the capabilities and limitations of the intended users. Human perception, information processing, memory and problem-solving are some of the concepts from cognitive psychology that are related to people's use of computers (Dix et al 2004).

We return to cognitive psychology and its role in HCI in lesson 2. In lesson 1, we explain the historical context within which HCI developed, the current context within which it is practiced, and we provide some definitions of "human-computer interaction" and related concepts.

1.2 The Historical Context

To put the development of interactive computing devices into context, we start this lesson with an overview of the history of computing and HCI.

1.2.1 The Middle Ages

The early history of computing can be traced back to the narrow aims of mathematicians, logicians and astronomers who had particular calculations that needed to be performed. The Persian astrologer Al-Kashi (1393–1449) built a device to calculate the conjunction of the planets. Records of this work survived and were transported to Europe, although the device itself was lost. The German mathematician Wilhelm Schickard (1592–1635) developed a much less sophisticated tool to perform simple addition and subtraction. The Schickard machine was destroyed during the Thirty Years' War. The French mathematician Blaise Pascal (1612–1662) replicated much of Schickard's work and only succeeded in building an even more simplified version of that machine.

There was no gradual improvement in our knowledge over time. War, famine and the plague interrupted the development of mechanical computing devices. This, combined with the primitive nature of the hardware, meant that user interfaces were almost non-existent. The systems were used by the people who built them. There was little or no incentive to improve HCI.

1.2.2 The Eighteenth and Nineteenth Centuries

The agricultural and industrial revolutions in Western Europe created the need for external markets and external sources of raw materials. This greatly increased the level of trade that was already conducted in commodities such as spices, gold and slaves. This, in turn, led to a rapid expansion in the merchant navies maintained by many countries. In the past the captains of these ships relied upon local knowledge and expertise. They always followed the same sea routes. As trade developed, this expertise became less important. As a result, there was an increasing need to produce accurate maps and navigation charts. These involved the calculation of precise distances and longitudes needed for navigation.

The demand for navigational aids fuelled the development of computing devices. Charles Babbage (1791–1871) was a British mathematician and inventor whose early attempts were funded by the Navy Board. As in previous centuries, his Difference Engine was designed to calculate a specific function (6th degree polynomials): a + bN + cN2 + dN3 + eN4 + fN5 + gN6.

This machine was never completed. Babbage's second machine, called the Analytical Engine, was a more general computer. This created the problem of how to supply the machine with its program. Punched cards were used and became perhaps the first solution to a user interface problem. The idea was so popular that this style of interaction dominated computer use for the next century.

1.2.3 The Early Twentieth Century

With the rise of mass production techniques on the east coast of the United States, economic pressures for trade increased. This had the effect of drawing migrants who were fleeing famine in Ireland and Scandinavia. The rapid influx of people caused severe problems for the United States government. They wanted to monitor this flow in order to avoid the introduction of epidemics from certain parts of the world. They also wanted to build a profile of the population for tax reasons.

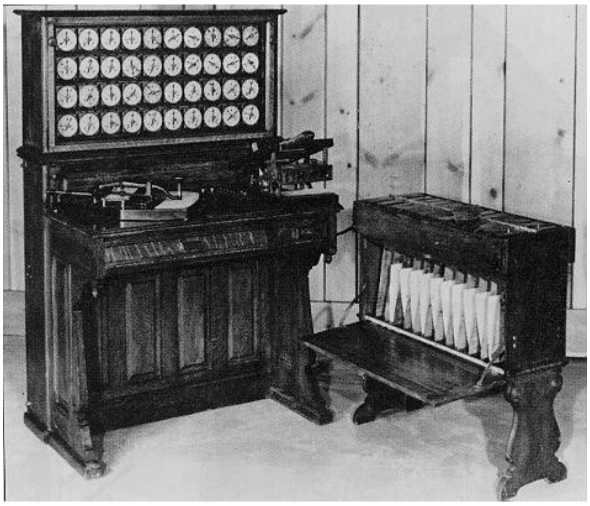


Figure 1.1: The Hollerith punch card and tabulating machine. From https://www.ibm.com/ibm/history/exhibits/vintage/vintage_products.html

As a result, Herman Hollerith (1860–1929) was recruited by the American census office to develop a computational device to calculate general statistics for the immigrant population. In 1887 he developed a punched card tabulating machine (see figure 1.1) that could sort more than 200 cards a minute. As a result, the 1890 census took about one-third of the time of the 1880 census.

These early attempts led to the foundation of the Computer-Tabulating-Recording Company (1911). This was possibly the first and the biggest computer company. In 1914 Thomas J Watson (Snr) joined the organisation and built it into the International Business Machines Corporation (IBM).

The important point here is that economic and political factors were intervening to create a greater market for computing devices. The term "computer" was originally used to describe the people who manually performed these calculations in the early twentieth century. In these early machines, the style of interaction was still based on the techniques pioneered in Babbage's Analytical Engine. Sequences of instruction were produced on punched cards. These were entered in batch mode, the jobs were prepared in advance, and interaction was minimal.

1.2.4 The Mid-Twentieth Century

The Second World War created another set of narrow applications for computing devices. Alan Turing, an English logician and a founder of computer science, was employed to break the German encryption techniques. This led to the development of the Colossus (1943), perhaps the first truly interactive computer. The operator could type input through a keyboard and gain output via a teleprinter.

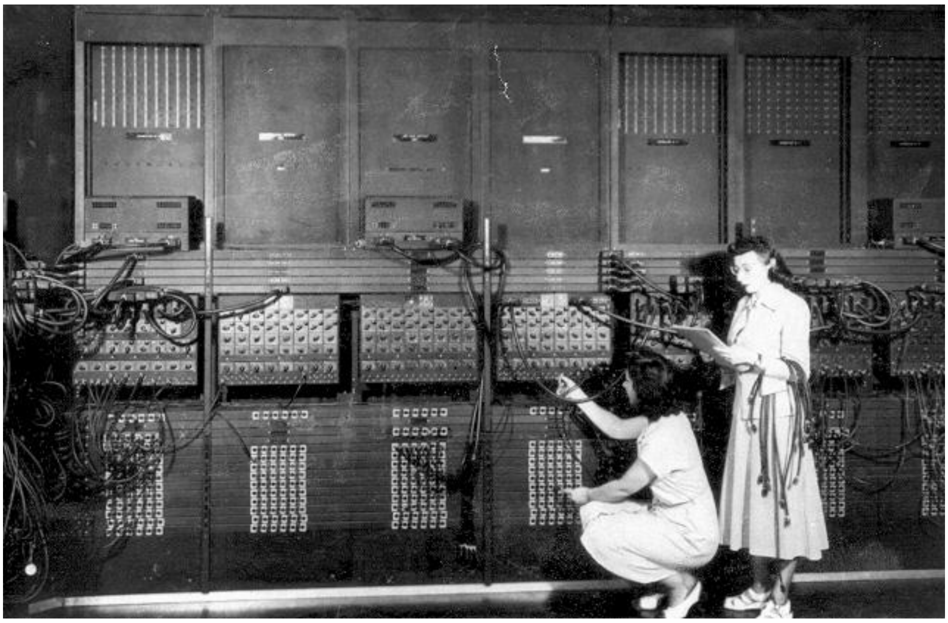


Figure 1.2: ENIAC (1943)

From http://ftp.arl.army.mil/~mike/comphist/ Two early programmers (Standing: Marlyn Wescoff Crouching: Ruth Lichterman) at work on the ENIAC US Army photo from the archives of the ARL Library (US Army Research Laboratory)

Many of the Colossus techniques were applied in the ENIAC machine (see figure 1.2), the first all-electronic digital computer produced around 1946 by JW Mauchly and JP Eckert in the United States. As with Colossus, the impetus for this work came from the military who were interested in ballistic calculations. To program the machine, you had to physically manipulate 200 plugs and 100 to 200 relays. The Manchester Mark I computer was also from about this period.

In 1945 Vannevar Bush, an electrical engineer in the USA, published his "As we may think" article in *Atlantic Monthly*. This article was the point of departure for Bush's idea of the Memex system. The Memex was a device in which individuals could store all personal books, records and communications, and from which items could be retrieved rapidly through indexing, keywords and cross-references. The user could annotate text with comments; construct a trail (chain of links) through the material and save it. Although the system was never implemented, and although the device was based on microfilm record rather than computers, it conceived the idea of hypertext and the World Wide Web (WWW) as we know it today.

By this time the first machine languages began to appear. These systems were intended to hide the details of the underlying hardware from programmers. In previous approaches, one was required to understand the physical machine. In 1957 IBM launched FORTRAN, one of the first high-level programming languages which created a new class of novice users: people who wanted to learn how to program but who did not want a detailed understanding of the underlying mechanisms. FORTRAN was based on algebra, grammar and syntax rules, and became the most widely used computer language for technical work.

In the early 1950s some of the earliest electronic computers such as MIT's Whirlwind and the SAGE air-defence command and control system, had displays as integral components. By the middle of the 1950s it became obvious that the computer could be used to manipulate pictures as well as numbers and text. Probably the most successful in this area was Ivan Sutherland who, in 1963, developed the SketchPad system at the MIT Lincoln Laboratory. It was a sophisticated drawing package which introduced many of the concepts found in today's interfaces such as the manipulation of objects using a light-pen, (including grabbing objects, moving them and changing their size) and using constraints and icons. Hardware developments that took place during the same period include "low-cost" graphics terminals, input devices such as data tablets, and display processors capable of real-time manipulation of images.

Two of the most dominant influences in suggesting the potential of the technology of this era have been Doug Engelbart and Ted Nelson. They both took the concept of the Memex system and elaborated on it in various ways. Whereas Nelson focused on links and interconnections (which he named 'hypertext' and implemented as the Xanadu system), Engelbart concentrated primarily on the hierarchic structure of documents. In 1963 he published an article entitled "A conceptual framework for augmenting human intellect", in which he viewed the computer as an instrument for augmenting man's intellect by increasing his capability to approach complex problem situations.

1.2.5 Turning Points

The turning points in the development of computers that would allow it to become available to the man in the street occurred in the mid-1970s – also the period which saw the rise of two major American role players in today's computer industry: Microsoft and Apple Computers. Initial attempts to support the "desktop metaphor" pushed graphical facilities and processor speeds to their limits. Below we follow the development towards where we are today.

1.2.5.1 Apple's First Personal Computer Kits (1976)

The Apple Company was founded by Steve Jobs and Steven Wozniak in 1976. Initially, they produced a series of kit machines similar to those that led to the development of the IBM PC a few years later. They hit upon the idea of pushing the code needed to represent the desktop into hardware. Graphics and device handling were burned into ROM (read-only memory). This led to a higher degree of consistency because it became more difficult to change the look and feel of the interface. (The Apple history website http://www.apple-history.com/ provides greater detail about their early history.)

Apple I (figure 1.3) was Steven Wozniak's first personal computer. It made its first public appearance in April 1976 at the Homebrew Computer Club in Palo Alto, but few took it seriously. It was sold as a kit one had to assemble. At $666, it was an expensive piece of machinery even by today's standards, considering that the price only included the circuit board. Users even had to build the computer case themselves. It was based on the MOStek 6502 chip (most other kit computers used the Intel 8080 chip).

Figure 1.3: Apple I From: http://en.wikipedia.org/w/index.php?title=Apple_I&oldid=499416018

1.2.5.2 IBM's First Personal Computer (1981)

Before the 1980s, personal computers were only used by enthusiasts. They were sold in kits and were distributed through magazines and electronic shops. This meant that their user population consisted almost entirely of experts. They understood the underlying hardware and software mechanisms because they had built most of it. Many people thought that they were mere toys. In the late seventies this attitude began to change as the demand for low-end systems began to increase.

Figure 1.4: IBM PC
From IBM Archives (www.ibm.com/ibm/history https://www.ibm.com/ibm/history/exhibits/vintage/vintage_products.html)

In 1981 IBM introduced their first PC (see figure 1.4) together with DOS (Disk Operating System). Little has changed in the underlying architecture of this system since its introduction. The relatively low cost and ease with which small-scale clusters could be built (even if they were not networked), vastly expanded the user population. A cycle commenced in which more people were introduced to computers. Increasing amounts of work were transferred to these systems and this forced yet more people to use the applications. As a result, casual users began to appear for the first time. They were people whose work occasionally required the use of a computer but who spend most of their working life away from a terminal. This user group found PCs hard to use. In particular, the textual language required to operate DOS was perceived to be complex and obscure.

1.2.5.3 The Xerox Star (1982)

Although the graphical user interface (GUI) had its roots in the 1950s, it was not developed until the 1970s when a group at the Xerox Palo Alto Research Center (PARC) developed the Alto, a GUI-based computer. The Alto was the size of a large desk, and Xerox believed it to be unmarketable. In 1982 Xerox introduced their STAR user interface. This marks what many people believe to be the beginning of HCI as a conscious design activity by software companies. In response to the increasing use of PCs by casual users and in office environments, Xerox began to explore more intuitive means of presenting the files, directories and devices that were represented by obscure pieces of text in DOS. Files were represented by icons and were deleted by dragging them over a wastebasket. Other features or principles included a small set of generic commands that could be used throughout the system, a high degree of consistency and simplicity, a limited amount of user tailor ability – what you see is what you get (WYSIWYG) – and the promotion of recognising/pointing rather than remembering/typing. It was the first system based upon usability engineering. Ben Shneiderman of the University of Maryland coined the term “direct manipulation” in 1982 and introduced the psychological foundations of computer use (Myers 1998). In the lesson tools that follow we will come back to the importance of some of these principles in interface design.

Steve Jobs of Apple Computers took a tour of PARC in 1979 and saw the future of personal computing in the Alto. Although much of the interface of both the Apple Lisa (1983) and the Apple MacIntosh (Mac) (1984) was based (at least intellectually) on the work done at PARC, much of the Mac OS (operating system) was written before Jobs’ visit to PARC. Many of the engineers from PARC later left to join Apple. When Jobs accused Bill Gates of Microsoft of stealing the GUI from Apple and using it in Windows 1.0, Gates fired back: “No, Steve, I think it’s more like we both have a rich neighbour named Xerox, and you broke in to steal the TV set, and you found out I’d been there first, and you said. ‘Hey that’s not fair! I wanted to steal the TV set!’”

The fact that both Apple and Microsoft got the idea of the GUI from Xerox put a major dent in Apple’s lawsuit against Microsoft over the GUI several years later. Although much of The Mac OS was original, it was similar enough to the old Alto GUI to make a look-and-feel suit against Microsoft doubtful. Today the look and feel of the Microsoft Windows Environment and the Macs are very similar, although both have retained some of their original unique features and identities (also in the naming of features). As far as hardware is concerned, the Apple and the PC have developed in more or less the same direction. The only difference is that Apple has experimented beyond pure functionality as far as the aesthetics of their machines is concerned. The Macintosh was the first popular computer to use a mouse and graphical user interface (GUI). The Macintosh was initially used as a desktop publishing tool.

Complete activity 1.1 (at end of the lesson)

1.2.5.4 The Internet, World Wide Web and Social Networks

Early in 1962 Rand Corporation, one of America’s leading military suppliers, became concerned about how people would communicate after a nuclear holocaust. Their solution (called ARPANET) was to grow into the internet – a highly connected network of computer systems. Since the inception of the internet, there has been a rapid growth in world-wide computer networks. In 1971 there were twenty-three host machines. In 1980 there were approximately one hundred computers attached to the internet and in 1990 one hundred thousand. In 1994 the number of systems connected to the internet exceeded one million. A 1999 estimate placed the number of internet users at more than two hundred million, in 2002 there were over five hundred million, and in 2009 there were 1.73 billion internet users worldwide.

Hundreds of sites in many different domains provide access to a vast range of information sources. The growth of these information sources and the development of applications such as Internet Explorer, Netscape and Mosaic, encouraged the active participation of new groups of users. Most of these participants possess only minimal knowledge of the communications mechanisms that support computer networks.

Two major developments based on the internet are the use of electronic mail systems (e-mail) and the World Wide Web (WWW). Lately, web-based social networks have emerged.

- **E-mail**

Until the late 1980s the growth in electronic mail was largely restricted to academic communities, in other words, universities and colleges. It then became increasingly common for companies to develop internal mail systems which were typically based around proprietary systems that were sold as part of a PC networking package. Most large businesses could not see the point of hooking up to the internet and so addresses were only valid within that local area network. Concerns over internet security also encouraged businesses to isolate their users' accounts from the outside world. But the situation has changed. The ability to transfer information rapidly using systems such as Microsoft's Internet Explorer and Netscape, has encouraged companies to extend their e-mail access. In 2009 close to 250 billion e-mails were being sent daily (http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/). Today users cannot function without the World Wide Web. In 2012 a total of 2.2 billion e-mail users and 144 billion e-mails per day have been reported worldwide.

- **The World Wide Web (WWW)**

The WWW originated at the National Centre for Supercomputer Applications (NCSA) at the University of Illinois and at CERN, the European Research Centre for Nuclear Physics, where there were concerted efforts to improve the means of passing files over remote networks. Application or client programs, called browsers, translate user requests for information into the communications primitives that are necessary to transfer relevant data from remote servers. The work at the NCSA led to the development of the Mosaic Web browser in 1993. Mosaic was a free program that did much to attract the initial user group to the Web. Netscape was then developed as a commercial successor once there were enough users for the Web to be successful. Users were reluctant to pay for a browser until there was enough information on the Web. Microsoft's Internet Explorer followed soon after.

An extensive user community has developed on the Web since its public introduction in 1991. In the early 1990s, the developers at CERN spread word of the Web's capabilities to scientific audiences worldwide. By September 1993 the share of Web traffic traversing the NSFNET Internet backbone reached 75 gigabytes per month or one per cent. By July 1994 it was one terabyte per month, and in the beginning of the 2000s it was in excess of ten terabytes a month. In 2009 there were more than 230 million websites and 1.73 billion internet users worldwide. Close to 250 billion e-mails were being sent daily by 2009 (http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/).

The World Wide Web, in short referred to as the Web, boasts different types of search engines, Wikipedia, the blogosphere, and a range of other systems such as microblogging platforms (e.g., Twitter), social network sites (e.g., Facebook), citizen science projects and human computation systems (e.g., Foldit) (Smart & Shadbolt 2018). It is clear that the Web played a very important role in the functioning of society, infrastructure (transport system) and industries in 2019. As Smart and Shadbolt (2018) indicate, the Web has managed to integrate itself into practically every form of social life. Few endeavours are undertaken without some sort of Web-based involvement. The Web is used not only for social life but also in technological processes and resources.

- **Social networks**

A social network is a social structure that connects individuals (or organisations). Connections are based on concepts such as friendship, kinship, common interest, financial exchange, dislike, sexual relationships or relationships of beliefs, knowledge or prestige. The WWW and mobile technology have become important platforms for new forms of social networks. The best-known examples are probably Facebook and Twitter.

Facebook is a social networking website that was started in February 2004 by Mark Zuckerberg (then 20 years old). Members of Facebook create personal profiles, photo albums and information walls to share happenings in their lives with people around the world. Facebook profiles are not exclusively for individuals. Schools, associations, companies, and so on can also have profiles and friends on Facebook. Anyone older than 12 can become a Facebook user and it costs nothing. In 2010 the website had more than 400 million active users worldwide. In 2019 it was reported that Facebook had 2.32 billion monthly active users and 1.15 billion mobile daily active users (https://zephoria.com).

Networks such as Facebook do not come without problems. Facebook has been banned in several countries because it is used to spread political propaganda. Many companies block their employees from accessing Facebook to prevent them from spending time on the network during working hours.

Twitter is a social networking and microblogging service that enables its users to communicate through tweets. It was created in 2006 by Jack Dorsey. Tweets are text-based messages (consisting of a maximum of 140 characters) that appear on the author's profile page for viewing by people subscribed to that page (they are known as the author's followers). Since late 2009 users are able to follow lists of authors instead of individual authors. Users can send and receive tweets via the Twitter website, external applications or a Short Message Service (SMS). Twitter had over 100 million users worldwide in 2010. According to Statista, the number of active Twitter users grew between 2010 and 2018 to 328 million per month, making Twitter the largest social network in the world (https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/).

Members of both Facebook and Twitter can either restrict the viewing of their information to users who are registered as their friends or followers, or they can allow open access.

1.2.5.5 Mobile Computing

Mobile computation can take place over great distances using cellular and satellite telephone links. It has made internet access an integral part of everyday life through devices such as notebook computers, personal digital assistants (PDAs) such as the iPhone®, and standard cellphones. Mobile computing is a set of IT technologies, products, services, operational strategies and procedures that enable users to gain access to computation, information and related resources while mobile. A mobile computing device is any device that is created using mobile components such as mobile hardware and software. It is portable and capable of operating, executing and providing services and applications similar to a typical computing device.

Mobile laptop and notebook computers can use one of two types of wireless access services when away from the home or office:

- WiFi uses radio waves to broadcast an internet signal from a wireless router to the immediate surrounding area. If the wireless network is not encrypted, anyone can use it. WiFi is commonly used in public places to create hotspots.
- Cellular broadband technology typically involves a cellular modem or card to connect to cell towers for internet access. The modem or card is inserted into a notebook computer when the user wants to access the internet.

Cellular broadband connections also make it possible to provide internet access through cellphones and PDAs. The latter depends on the model of phone and on the type of contract with the service provider.

1.3 The Current Context and Future Directions

1.3.1 The Current Context

The following aspects of computer use currently affect HCI:

- **Distributed systems**: The development of innovative user interfaces is increasing access to distributed information sources. People surfing the net are no longer just programmers looking for interesting pieces of code. The term "Web 2.0" refers to web applications that make interactive information sharing and collaboration on the World Wide Web possible. Users now contribute to the content of the website whereas first-generation websites were limited to passive viewing of information. Facebook (discussed above) is an example of a Web 2.0 website. The next generation web applications (Web 3.0) involve web-based applications that talk to one another to produce new information. Content is thus generated by the web applications themselves.

- **Multimedia interfaces**: Text is still the most significant form of interaction with computer systems. However, integrating it into graphical, video and audio information sources poses a problem.

- **Advanced operating systems**: Many of the changes described in section 1.2 have been driven by changes in the underlying computer architecture. Increasing demands are made upon processing resources by graphical and multimedia styles of interaction. These demands are being met by the improvement of operating systems such as OS2 and Windows 2010 which allows for much improved manipulation of multimedia documents.

- **HCI development environments**: On top of the new generations of operating systems, there are even newer generations of interface development software. Many of these environments extend the graphical interaction techniques of the Apple and Windows desktops to the construction of the interface itself. For perhaps the first time, users may be able to customise their working environment. This creates opportunities but also carries high risks if different users have to operate the same application at different times.

- **Ubiquitous computing (UbiComp)**: This refers to computer systems that are embedded in everyday objects and have unobtrusively become part of the environment. An example is the computerised control systems found in modern cars (that activate, for example, windshield wipers at the appropriate wiping speed when rain is detected or when your car lights go on when entering a darker area).

- **Mobile technology**: This has changed the context within which technology is used, the composition of the user population, as well as the design of user interfaces. Computers (in their mobile form) can be used any time any place. Through mobile technology, people who would never have access to computers in their non-mobile form, now have mobile phones through which they can access resources such as the WWW. What could previously only be done on a desktop PC, can now be done on mobile phones or devices. In 2007 about 77% of all Africans had mobile phones, whereas only 11% had computer access. To support this market, designers have to find ways to create user interfaces that fit into the small displays of mobile devices. In 2017 a GSMA report indicated that 5 billion people worldwide had a mobile phone connection (https://venturebeat.com). Statistics have also shown that between 2015 and 2020 approximately 20 million people in South Africa were using smartphones (https://www.statistica.com). The number of people having mobile connections is much higher (90 million) whereas the number of people having access stands at 80 million. The focus in HCI has moved beyond the desktop to accommodate hand-held devices.

- **Gaming:** Computer and video games are the most popular and important products of the software industry. A group called HCI Games conducts research in ICT, design, psychology and HCI-related areas.

1.3.2 Future Directions

Mobile and ubiquitous computing will remain the focus areas of the future. Harper, Rodden, Rogers and Sellen (2008) have identified five major transformations in computing that will affect the field of HCI increasingly in the next decade. These are:

1. **The changing notion of "the interface"**
   Outdated ideas of what an interface is will not apply in the era of ubiquitous computing. Graphical user interfaces and the mouse will increasingly be replaced by tangible interfaces controlled by touch, speech and gesture and input mechanisms using eye movement and brain activity will become more commonplace. Embedded devices have no explicit interface, which places interface design in a completely different perspective.

2. **Increasing dependency on technology**
   For older people who have grown up without the internet and mobile phones it is still possible to imagine life without them. But younger generations who have always had these instantly available will not be able to function without them. Although users are constantly becoming more knowledgeable about the use of technology, they are also losing skills that previous generations had because they did not have constant access to technology (e.g., mental calculation and possibly memory skills).

3. **Hyper connectivity**
   Communication technology will continue to improve and allow even more forms of connectivity among people. The rapid growth in connectivity will impact on the way we relate to people, how we make friends and how we maintain relationships. The etiquette of when, how and with whom we communicate, is also changing. For example, students send e-mails to their lecturers using the same slang they would use with their friends, whereas, in person, they would never speak that way to the lecturer. People engage in romantic relationships with someone whom they have never met face to face. Where previously there was a clear distinction between workspace (or time) and leisure space (or time), the levels of connectivity have blurred these boundaries. The question is now: What will the effect of this be on our social make-up in the long run?

4. **Changes in the means of and reasons for recording information**
   Information that was previously stored in people's memories are now recorded in digital format. Mobile phones with cameras and video recording capabilities allow people to record activities that would previously be forgotten. The replacement of handwritten letters by e-mails means that it is much easier to build archives of everyday communications. The increase in recorded information requires improved systems for managing storage and accessing information.

5. **Increased creativity through technology**
   Increasingly, accessible and flexible computing devices can support new ways of playing, learning and creating. They become tools that can augment human cognition by visualising

complex data or processes or by processing huge amounts of information in short periods of time.

According to Shneiderman, Plasant, Cohen and Jacobs (2014), it is important to look at not only mobile and ubiquitous computing but also hardware and software diversity. They identified three technical challenges for the next decade:

1. **Producing satisfying and effective internet interaction on high-speed (broadband) and slower (dial-up and some wireless) connections**.
   Although a great deal of research has been conducted to reduce the file size of images, music, animation and even videos, more needs to be done. Newer technologies need to be developed to enable pre-fetching or scheduled downloads.
2. **Enabling access to web services on large displays (1200x 1600 pixels or larger) and small mobile devices (640 x 480 and smaller).**
   Designers need to design web pages for different display sizes to produce the best quality, which can be costly and time-consuming for web providers. New software tools are needed to allow website designers to specify their content in a way that enables automatic conversations for an increasing range of display sizes.
3. **Supporting easy maintenance of or automatic conversation to multiple languages.**
   Commercial companies realise that they can expand their markets if they can provide access in multiple languages and across various countries.

Complete activity 1.2 (at end of lesson)

1.4 HCI and Related Concepts and Fields

HCI emerged in the early 1980s as a specialty area in computer science, and since then has developed as an area of research and practice that attracts professionals from a wide range of disciplines (Carroll 2009).

1.4.1 Definitions of HCI

Various definitions for the term “human-computer interaction” have been put forward over the years. Here are a few:

- HCI is a “set of processes, dialogues, and actions through which a human user employs and interacts with a computer” (Baecker & Buxton 1987).

- HCI is a “discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them. From a Computer Science perspective, the focus is on interaction and specifically on interaction between one or more humans and one or more computational machines” (ACM SIGCHI, [sa]). A computational machine (computer) is defined to include traditional workstations as well as embedded computational devices such as spacecraft cockpits or microwave ovens, and specialised boxes such as electronic games. A human is defined to include a range of people, from children to the elderly, computer aficionados to computer despisers, frequent users to hesitant users, big-hulking teenagers to people with special needs.

- HCI is “the study of people, computer technology, and the ways these influence each other” (Dix et al 2004). A (human) user is defined as whoever tries to accomplish something using technology and can mean an individual user, a group of users working together, or a sequence of users in an organisation, each dealing with some part of the task or process. A computer is defined as any technology ranging from a general desktop computer to large-scale computer systems, a process control system or an embedded system. The system may include non-computerised parts, including other people. Interaction is defined as any communication between a user and a computer, be it direct or indirect. Direct interaction involves a dialogue with feedback and control during performance of the task. Indirect interaction may involve background or batch processing.

- HCI is concerned with studying and improving the many factors that influence the effectiveness and efficiency of computer use. It combines techniques from psychology, sociology, physiology, engineering, computer science and linguistics (Johnson 1997).

There are several other terms and fields of study that have strong connections with HCI. Some are listed below:

- *Ergonomics* is the study of work. The term "ergonomics" is widely used in the United Kingdom and Europe in contrast to the United States and the Pacific basin where the term "human factor" is more popular (see below). Ergonomics has traditionally involved the design of the "total working environment" such as the height of a chair and desk. Health and safety legislations such as the UK Display Screen Equipment Regulations (1992), blur the distinction between HCI and ergonomics. In order to design effective user interfaces, we must consider wider working practices. For instance, the design of a telesales system must consider the interaction between the computer application, the telephone equipment and any additional paper documentation.

- The *human factor* is a term used to describe the study of user interfaces in their working context. It addresses the entire person and includes:

    - physiology: our physical characteristics such as height and reach
    - perception: our ability to sense information by hearing, touching and seeing it
    - cognition: the way we process data such as the information we extract from a display.

  It has much in common with ergonomics but is often used to refer to HCI in the context of safety-critical applications. Physiological problems have a greater potential for disaster in these systems.

- *Usability* is defined by the International Standards Organisation (ISO) as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use". The ISO standard 9241 gives the following definition of its components:

    - Effectiveness: The accuracy and completeness with which specified users can achieve specified goals in particular environments.
    - Efficiency: The resources expended in relation to the accuracy and completeness of goals achieved.
    - Satisfaction: The comfort and acceptability of the work system to its users and other people affected by its use.

- *User experience* refers to how people feel about a product. How satisfied are they when using it, looking at it or handling it? It includes the overall impression as well as the small effects such as how good it feels to touch. According to Preece, Rogers and Sharp (2019), you cannot design a user experience; you can only design *for* user experience.

- *Interaction design* is defined by Preece et al (2019:xvii) as "designing interactive products to support the way people communicate and interact in their everyday and working lives". It involves four activities, namely:

  - identifying needs and establishing user requirements
  - developing alternative designs according to the requirements
  - building prototypes of the designs so that they can be assessed
  - evaluating the designs and the user experience.

- *Accessibility*, in the context of HCI, means "designing products so that people with disabilities can use them. Accessibility makes user interfaces perceivable, operable, and understandable by people with a wide range of abilities, and people in a wide range of circumstances, environments, and conditions. Thus accessibility also benefits people without disabilities, and organizations that develop accessible products" (Henry 2007). It is the degree to which a system is usable by people with disabilities (Preece at al 2019). Some people see accessibility as a subset of usability; others regard it is a prerequisite for usability.

Complete activity 1.3

1.4.2 Who is Involved in HCI?

HCI is a multidisciplinary or interdisciplinary field of study. The ideal designer of interactive systems should have expertise in a variety of topics (Dix et al 2004, Preece et al 2019), including:

- Psychology and cognitive science: give insight into the user's capabilities and perceptual, cognitive and problem-solving skills.
- Environmental factors and ergonomics: to be able to address the user's working environment, physical capabilities and comfort factors.
- Organisational factors: to be able to address training, job design, productivity and work organisation.
- Health and safety factors.
- Philosophy, sociology and anthropology: help to understand the wider context of interaction.
- Linguistics.
- Computer science and engineering: to be able to build the necessary technology.
- Graphics design: to produce effective interface presentation.

No person, not even the average design team, has so much expertise. In practice, designers tend to be strong in one aspect or another. It is, however, not possible to design effective interactive systems from one discipline in isolation. There is a definite interaction among all these disciplines in designing and developing an interactive artefact.

Professionals in HCI are widely spread across the spectrum of subfields. They are, for example, user experience designers, interaction designers, user interface designers, application designers, usability engineers, user interface developers, application developers or online information designers (Carroll 2009).

Complete activity 1.4

1.5 Conclusion

The purpose of studying human-computer interaction is to improve the quality of the interaction between human and machine by systematically applying knowledge about human capabilities and limitations and machine capabilities and limitations. It is also to improve the productivity, functionality, effectiveness, efficiency and usability of technology. Human-computer interaction is about designing for people (users). It is, however, not only about the display and the keyboard or about people in offices. It has a much wider context and addresses interactive situations in everyday life as well. Therefore, although the main purpose of this module is to introduce you to HCI, the aim is also to create an awareness of user-centred design in general. User-centred design is not easy but will increase in importance with the changes taking place in technology.

1.6 Activities

ACTIVITY 1.1

Complete the timeline in table 1 for the historical context of human-computer interaction by supplying the missing information.

Table 1: The historical context of human-computer interaction

< 1450	The Persian astrologer ________ used a device to calculate the conjunction of planets.
1600	The German mathematician ________ developed a tool to perform simple addition and subtraction.
	Blaise Pascal built a simplified replica of ________'s device.
1820 – 1870	Charles Babbage built his ________ to calculate 6^{th}-degree polynomials.
	________ developed the Analytical Engine which was the first calculating device to use punched cards.
________	Herman Hollerith developed a computational device using punched cards to calculate census statistics.
1914	Thomas J Watson joined the ________ Company and built it up to form the International Business Machines Corporation (IBM).
1943	________ developed the ________ to break German encryption techniques. The Colossus accepted input via a keyboard and produced output via a teleprinter.

________	Vannevar Bush published an article entitled "As we may think" in *Atlantic Monthly* introducing his Memex system.
1946	The ________ machine, the first all-electronic digital computer, was produced by JW Mauchly and JP Eckert in the United States.
________	IBM introduced the FORTRAN high-level programming language.
1963	Ivan Sutherland developed the ________ system at the MIT Lincoln Laboratory. It was the first sophisticated drawing package.
1976	Steven Wozniak produced Apple I, based on the ________ Chip.
________	IBM produced their first PC with DOS.
1982	Xerox produced their ________ in which files were represented by icons and were deleted by dragging them over a wastebasket. This marked the advent of the modern desktop.

ACTIVITY 1.2

Name and describe one specific example of each of the following:

- a mobile device
- a web-based social network (excluding Facebook and Twitter)
- a ubiquitous computing device

ACTIVITY 1.3

Define the following terms:

- human-computer interaction
- ergonomics
- usability
- interaction design
- accessibility

ACTIVITY 1.4

Do your own research on the internet to find out what each of the occupations below entail. Then, assuming that you are the CEO of a dynamic new web application development company, formulate job advertisements for each of the positions. Include the required qualifications and experience as well as the key tasks that the person will perform.

- usability engineer
- interaction designer
- user experience designer

2.2 Lesson 2: Human Issues in HCI

The contents of this lesson are as follows:

CONTENTS

LESSON 2 – OUTCOMES

After studying this lesson tool, you should:

- understand how perception and cognition influence user interface design
- understand the differences between short- and long-term memory and their influence on interface design
- know how the physical attributes of users can affect their interaction with computers
- understand how users may differ in terms of culture, personality, gender and age, and how these differences impact on HCI
- understand how different levels of expertise influence users' interaction with a system
- know the kinds of errors humans make, why they make them, and how this should be addressed in user interface design

2.1 Introduction

In this lesson, we focus on the "human" in human-computer interaction. We address some of the differences in and between user populations that must be considered when developing and installing computer systems. In particular, we will identify the effects that perception, cognition and physiology can have on human performance. We will also touch on the issues of personality and cultural diversity and will discuss the special needs and characteristics of users in different age groups.

Part of human nature is to make errors. We look at the different kinds of errors people make and discuss ways to avoid them.

Not all information will be relevant to every commercial application. For instance, the developers of a mass market database system may have little or no control over the workstation layout of their users. In other contexts, particularly if you are asked to install equipment within your own organisation, these factors are under your personal control. It is important that, as future software designers and information technology managers, you are made aware of the factors that influence people's experience with technology.

2.2 Cognitive Psychology in HCI

Many cognitive processes underlie the performance of a task or action performed by humans. Human information processing consists of three interacting systems: the perceptual system, the cognitive system, and the motor system. We can therefore characterise human (user) resources into three categories:

- Perception: the way that people detect information in their environment.
- Cognition: the way that they process that information.
- Physiology: the way in which they move and interact with physical objects in their environment.

A vital foundation for designers of interactive systems is an understanding of the cognitive and perceptual abilities of the user. Some regard perception as part of cognition (Preece et al 2007; Preece et al 2019), but here we will discuss it as a separate aspect of human information processing.

2.2.1 Perception

Perception involves the use of our senses to detect information. The human ability to interpret sensory input rapidly and initiate complex actions makes the use of modern computer systems possible. In computerised systems, this mainly involves using the senses to detect audio instructions and output, visual displays and output, and tactile (touchable) feedback.

Information from the external world is initially registered by the modality-specific sensory stores or memories for visual, audio and tactile information respectively. These stores can be regarded as input buffers holding a direct representation of sensory information. But the information persists there for only a few tenths of a second. So, if a person does not act on sensory input immediately, it will not have any effect.

Shneiderman et al (2014:71) identified several design implications for the design of information to be perceptible and recognisable across different media:

- Icons and other graphical representations should enable users to readily distinguish their meaning.
- Boarders and spacing are effective visual ways of grouping information and make it easier to perceive and locate items.
- If sound is used, it should be audible and distinguishable so that users understand what they represent.
- Speech output should enable users to distinguish between sets of spoken words and to understand what they mean.
- Text should also be legible and distinguishable from the background.
- When tactile feedback is used in a virtual environment, it should allow users to recognise the meaning of the touch sensations being emulated, for example, the sensation of squeezing is represented in a tactile form that is different from the sensation of pushing.

Many factors affect perception, for example:

- A change in output such as changes in the loudness of audio feedback or in the size of elements of the display.
- Maximum and minimum detectable levels of, for example, sound. People hear different frequencies. They also differ in the number of signals they can process at a time.
- The field of perception. Depending on the environment, not all stimuli may be detectable. Not all parts of the display may be visible if a user, for example, faces it at the wrong angle.
- Fatigue and circadian (biological) rhythms. When people are tired, their reactions to stimuli may be slower.
- Background noise.

Designers have to make sure that people can see or hear displays if they are to use them. In some environments, this is particularly important. For instance, most aircraft produce over 15 audible warnings. It is relatively easy to confuse them under stress and with high levels of background noise. Such observations may be worrying for the air traveller, but they also have significance for more general HCI design. We must ensure that signals are redundant (e.g., it must be more than what is needed, desired or required). If we display critical information through small changes to the screen, many people will not detect the change. If you rely upon audio signals to inform users about critical events, you exclude people with hearing problems or people who work in a noisy environment. On the other hand, audio signals may irritate users in shared offices.

Partial sight, ageing and congenital colour defects produce changes in perception that reduce the visual effectiveness of certain colour combinations. Two colours that contrast sharply when perceived by someone with normal vision may be far less distinguishable to someone with a visual defect. People with colour perception defects generally see less contrast between colours than someone with normal vision. Lightening light colours and darkening dark colours will increase the visual accessibility of a design.

Three aspects of colour influence how they are perceived:

- Colour *hue* describes the perceptual attributes associated with elementary colour names. Hue enables us to identify basic colours such as blue, green, yellow, red and purple. People with normal colour vision report that hues follow a natural sequence based on their similarity to one another.
- Colour *lightness* corresponds to how much light is reflected from a surface in relation to nearby surfaces. Lightness, like hue, is a perceptual attribute that cannot be computed from physical measurements alone. It is the most important attribute in making contrast more effective.
- Colour *saturation* indicates a colour's perceptual difference from a white, black or grey of equal lightness. Slate blue is an example of a desaturated colour because it is similar to grey.

Congenital and acquired colour defects make it difficult to discriminate between colours on the basis of hue, lightness or saturation. Designers can compensate for these defects by using colours that differ more noticeably with respect to all three attributes.

Complete activity 2.1

2.2.2 Cognition

Cognition refers to a variety of processes that take place in our heads. These include:

- short-term memory and information processing
- long-term memory and learning
- problem-solving
- decision-making
- attention
- search and scanning
- time perception

Knowledge of these will help designers to create usable interfaces. Our discussion of cognition will be limited to attention and memory.

2.2.2.1 Attention

Attention is the process of concentrating on something (e.g., an object, a task or a conversation) at a specific point in time. It can involve our senses such as looking at the road while driving or listening to a news story on the radio, or it can involve thinking processes such as concentrating on solving a mathematical problem in your head. People differ in terms of their attention span. Some people are distracted easily whereas others can concentrate on a task in spite of external disturbances. In the past 10 years it has become even more common for people to switch between multiple tasks (Shneiderman et al 2014). Attention allows us to focus on information which is relevant to what we are doing.

Attention is influenced by the way information is presented as well as by people's goals (Preece et al 2007, 2019). This has implications for designers of computer systems. If information on an interface is poorly structured, users will have difficulty in finding specific information. How information is displayed determines how well people will be able to perform a searching task. When using a system with a particular goal in mind, the user's attention will remain focused more easily than when he or she aimlessly browses through an application. Designers of browsing and searching software should therefore find ways to lead users to the information they want. In a computer game, it is important that users always know what their next goal in the game is, otherwise they will lose interest.

The following activity is a good example of focusing your attention. Find the price of a family room in a guest house that has five rooms in table 2.1 (a). Then find the telephone number in table 2.1 (b). Which took longer to find information, table 2.1 (a) or table 2.1 (b)?

Table 2.1: Finding information relating to accommodation

Gauteng					
City	**Guest House**	**Area code**	**Phone**	**Rates**	
				Single	Double
Johannesburg	All-in-one	011	670 9232	R450	R900
All-in-one	Rest retreat	011	670 9232	R670	R1300
Rest retreat	Break a way	011	678 9834	R300	R600
Pretoria	All-in-one	012	690 1232	R550	R1000
Pretoria	Rest retreat	012	690 1453	R770	R1400
Pretoria	Break a way	012	678 9566	R330	R700

(a)

Cape Province
Holiday Inn Hotel: Cape Town
021 689 0987 S: R550 D: R1 200
Holiday Inn Hotel: Durbanville
021 689 1988 S: R450 D: R1 100
Holiday Inn Hotel: Muisenberg
021 6693458 S: R666 D: R1 499
Holiday Inn Hotel: Simon's Town
021) 679 2388 S: R455 D: R1 222
Protea Hotel: Cape Town
021 689 8743 S: R350 D: R780
Protea Hotel: Stellenbosch
021 787 8932 S: R456 D: R876

(b)

In early studies conducted by Tullis, it was found that the two screens produce different results: it takes on average 3,2 seconds to search for the information in table 2.1 (a) and 5,5 seconds to find the same kind of information in table 2.1 (b). The question one can then ask is: Why so? The primary reason is the way in which the characters are grouped in the display. In table 2.1 (a) the characters are grouped into vertical categories of information with columns of space between them. Because the information in table 2.1 (b) is bunched together, it is much harder to go through it.

Shneiderman et al (2014) identified a few guidelines which designers can use to get the user's attention with the precaution that it should be used moderately to avoid clutter:

- *Intensity*. The designer should make use of two levels only with a limited use of high intensity to draw attention.
- *Marking*. Underline an item, enclose it in a box, point at it with an arrow, or make use of an indicator such as an asterisk, bullet, plus sign or an X.
- *Size*. Use only four sizes; the larger sizes attracting attention.
- *Choice of fonts*. Never use more than three fonts.
- *Inverse video*. Use inverse colouring.
- *Blinking*. Make use of blinking display (2-4 Hz) or blinking colour changes but use with caution and only in limited areas.
- *Colour*. Use only four standard colours and reserve additional colours for occasional use.
- *Audio*. Use soft tones for regular positive feedback and harsh sounds for emergency conditions.

Audio tones such as the click of a keyboard or the ring tone of a telephone, can provide informative feedback about progress. Alarms that go off in an emergency are a good example of getting a user's attention, but there should also be a mechanism for the user to suppress alarms. An alternative to alarms is voice messages.

2.2.2.2 Memory

Memory consists of a number of systems that can be distinguished in terms of their cognitive structure as well as their respective roles in the cognitive process (Gathercole 2002). Authors have different views on how memory is structured, but most distinguish between long-term and short-term memories. Short-term memory (STM) stores information or events from the immediate past and retrieval is measured in seconds or sometimes minutes (Gathercole 2002). Long-term memory (LTM) holds information about events that happened hours, days, months or years ago and the information is usually incomplete.

STM has a relatively short retention period and is limited in the amount of information that it can keep. It is easy to retrieve information from STM. Some people refer to STM as "working memory" since it acts as a temporary memory that is necessary to perform our everyday activities. The effectiveness of STM is influenced by attention – any distraction can cause information to vanish from STM. Generally, people can keep up to seven items (e.g., a seven-digit telephone number) in their STM unless there is some distraction.

LTM, on the other hand, has a high capacity. As its name suggests, it can store information over much longer periods of time, but access is much slower. It also takes time to record memories there. If we have to extract the information from LTM, it may involve several moments of thought: for example, naming the seven dwarfs or the current members of the national soccer team. The information stored in LTM is affected by people's interpretation of the events or contexts. Information retrieved from LTM is also influenced by the retriever's current context or state of mind.

We should design interfaces that make efficient use of users' short-term memory. Users should be required to keep only a few items of information in their STM at any point during interaction. They should not be compelled to search back through dim and distant memories of training programmes in order to operate the system. User interfaces can support short-term memory by including cues on the display. This is effectively what a menu does: it provides fast access to a list of commands that do not have to be remembered. On the other hand, help facilities are more like long-term memory. We have to retrieve them and search through them to find the information that we need.

In line with the general STM capacity, seven is often regarded as the magic number in HCI. Important information is kept within the seven-item boundary. Additional information can be held, but only if users employ techniques such as chunking. This involves the grouping of information into meaningful sections. National telephone numbers are usually divided in this way: 012 429 6122. Chunking can be applied to menus through separator lines or cascading menus.

As we have mentioned, it takes effort to hold things in STM. We all experience a sense of relief when it is freed up. As a result of the strain of maintaining STM, users often hurry to finish some tasks. They want to experience the sense of relief when they achieve their objective. This haste can lead to error. Some ATMs issue money before returning the user's card. Users experience a sense of closure when they have satisfied their objective of withdrawing money. They then walk away and leave their cards in the machine. To avoid this, most ATMs dispense cash only after the user has removed the card. The computerised system is designed so as to prevent errors caused by the limitations of STM.

An important aim for user interface design is to reduce the load on STM. We can do this by placing information "in the world" instead of expecting users to have it "in the head" (Norman 1999). In computer use, knowledge in the world is provided through the use of prompts on the display and the provision of paper documentation.

Shneiderman et al (2014) indicated that users increasingly save their digital content on the Cloud, iCloud, Vimeo, Pinterest and Flickr so that they can access it from multiple platforms. The challenge these companies face is to provide interfaces that will enable users to store their content so that they can readily access specific items, for example, a particular image, video or document. In order to help users to remember what they saved, where they saved it or how they named the file, different recall methods are used. Initially, the user tries recall-directed memory and when it fails, recognition-based scanning, which takes longer. Designers should consider both kinds of memory processes so that users can use whatever memory they have to limit the area being searched and then represent the information in this area of interface.

Complete activity 2.2

2.2.2.3 Knowledge in the World vs Knowledge in the Head

Norman (1999) refers to information kept in someone's memory as "knowledge in the head" and to external information as "knowledge in the world". Both these kinds of information are necessary for our functioning in the world (and also for our interaction with computers), but how it is used depends on the individual. Some people rely more on knowledge in the world (e.g., notes, lists and birthday calendars) whereas others depend more on the knowledge in their heads (their memory). There are advantages and disadvantages to both approaches. These are summarised in table 2.2.

Table 2.2: Comparison of knowledge in the head and in the world (from Norman (1999))

Property	Knowledge in the world	Knowledge in the head
Retrievability	Easily retrievable whenever visible or audible. (Depends on availability in the environment.)	More difficult to retrieve. Requires memory search or reminding.
Learning	Learning is not required, only interpretation.	To get information there requires learning, which can be considerable.
Efficiency of use	Tends to be slowed up by the need to find and interpret the external sources.	Can be very efficient.
Ease of use at first encounter	High	Low
Aesthetics	Can be unesthetic and inelegant, especially if there is a need to maintain a great amount of information. Can lead to clutter. Requires a skilled designer.	Nothing needs to be visible, which gives the designer more freedom.

When designing interfaces, the trade-off between knowledge in the world and knowledge in the head must be kept in mind. Do not rely too much on the user's memory, but don't clutter the interface with memory cues or information that is not really necessary. Meaningful icons and menus can be used to relieve the strain on memory, but the Help menu should provide additional information "in the world" that is difficult to display properly on the interface.

Complete activity 2.3

2.2.2.4 Examples to Illustrate the Role of Memory in HCI

We end this section on cognition with two examples of how interface design can relate to human cognition (the user's memory in particular).

The image given in figure 2.1 is of a message from Microsoft's Word 97. The message appeared after you have spell-checked a document that contained text that you have indicated should be excluded from spell-checking (the no-proofing option). The message is certainly informative but requires that the user either has an exceptional short-term memory or has pen and paper handy to write down the steps that it refers to.

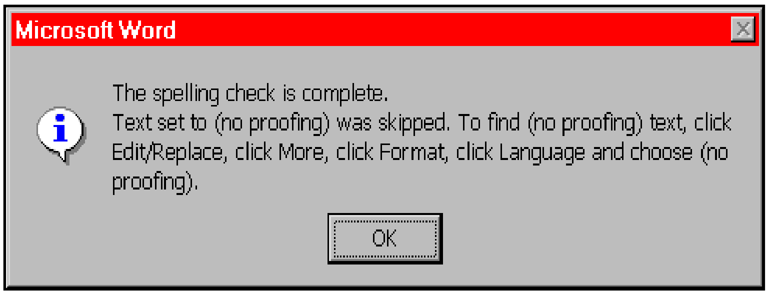


Figure 2.1 Proofing in Microsoft Word (Isys, 2000) From http://hallofshame.gp.co.at/mdesign.htm

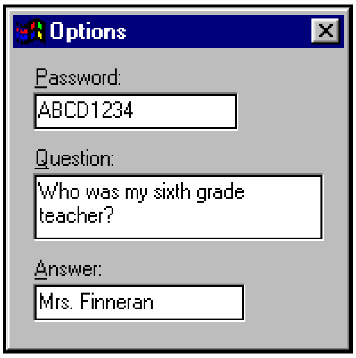


Figure 2.2: Mechanism to retrieve a forgotten password (Isys, 2000) From http://hallofshame.gp.co.at/mdesign.htm

Given all the passwords each of us must keep track of, it's all too easy to forget the password for a particular account or program. Figure 2.2 shows how many applications nowadays help the user to remember a password. When creating a new account, you are asked to specify the new password and, in addition, provide a question and answer in the event that you forget your password at some later time. The log-in window includes a "Forgot my password" button that will prompt you with the question you provided at registration and await your response.

This is a good solution to a problem that has plagued system operators everywhere. It is an interface feature that should be considered for every application that requires a password.

2.3 Physiology

Physiology involves the study of the human anatomy. It might seem strange to include this in a course on user interface design, but knowledge of physiology can make a noticeable contribution to the design of a successful system.

2.3.1 Physical Interaction and the Environment

When using a computer system, users must at least be able to view the interface and reach the input devices. Designers often have relatively little influence on the working environments of their users. If they do have some power, here are a few guidelines they can follow:

- Visual displays should always be positioned at the correct visual angle to the user. Even relatively short periods of rotation of the neck can lead to long periods of pain in the shoulders and lower back.
- Keyboard and mouse use: Prolonged periods of data entry place heavy stress upon the wrist and upper arm. A range of low-cost wrist supports are now available. They are a lot cheaper than the expense of employing and re-training new members of staff. Problems in this regard include repetitive strain injury and carpal-tunnel syndrome (both cause pain and numbness in the arms). Frequent breaks can help to reduce the likelihood of these conditions.
- Chairs and office furniture: It's no good providing a really good user interface if your

employees spend most of their time at a chiropractor. It is worth investing in well-designed chairs that provide proper lower back support and promote a good posture in front of a computer.

- Placement of work materials: Finally, it is important that users are able to operate their system in conjunction with other sources of information and documentation. Repeated gaze transfers lead to neck and back problems. Paper and book stands can reduce this.
- Other people: You cannot rely on system operators to prevent bad things from happening. Unexpected events in the environment can create the potential for disaster. For example, a patient monitoring system should not rely on a touch screen if doctors or nursing staff who move around the patient can accidentally brush against it.

It also pays to consider the possible sources of distraction in the working environment:

- Noise: Distraction can be caused by the sounds made by other workers (their phone calls or the buzz of their computers) and by office equipment (fans or printers). There are a number of low-cost solutions. For example, you may introduce screens around desks or covers for devices such as printers. High-cost solutions involve the use of white noise to mask intermittent beeps.
- Light: Bright lighting can distract users in their interaction with computers. Its impact can be reduced by blinds and artificial lighting to reduce glare in the room. A side-effect of this is that, over time, users may suffer from fatigue and drowsiness. Many Japanese firms have invested in high-intensity lighting systems to avoid this problem. Low-cost solutions involve moving furniture or using polarising filters.

There are also a number of urban myths (untruths) about the impact of computer systems on human physiology:

- Eyesight: Computer use does not damage your eyesight. It may, however, make you aware of existing defects.
- Epilepsy: Computer use does not appear to induce epileptic attacks. Television may trigger photosensitive epilepsy, but the visual display units of computers do not seem to have the same effect. The effect of multimedia video systems upon this illness is still unclear.
- Radiation: The National Radiological Protection Board in the UK stated that VDUs do not significantly increase the risk of radiation-related illnesses.

Interfaces often reflect the assumptions that their designers make about the physiological characteristics of their users. Buttons are designed so that an average user can easily select them with a mouse, touchpad or tracker-ball. Unfortunately, there is no such thing as an average user. Some users have the physiological capacity to make fine-grained selections, but others do not. Although users may have the physical ability to use these interfaces, workplace pressures may reduce their physiological ability.

A rule of thumb is: Do not make interface objects so small that they cannot be selected by a user in a hurry; also, do not make disastrous options so easy to select that they can be started by accident.

Complete activity 2.4

2.3.2 Users with Disabilities

Preece et al (2007) define accessibility as “the degree to which an interactive product is usable by people with disabilities” (p 483). There is a wide range of disabilities, including severe physical conditions such as blindness, deafness and paralysis, and less severe ones such as dyslexia and colour blindness. Then there are mental disabilities such as Down syndrome, autism and dementia. In the United States, more than forty-eight million people were disabled in 2006 (Kraus, Stoddard & Gilmartin 2006). The 2001 South African census revealed that more than two million people had some form of disability (Lehohla 2005). It was estimated that in 2006 more than five hundred million people around the world were disabled (United Nations 2006).

The statistics above provide ample reason to compel designers to take accessibility into consideration. It will have a profound impact on the development of the user interface if people with disabilities form part of the target market. Henry (2002) lists more reasons for designing systems that are accessible to people with disabilities. These include:

- Compliance with regulatory and legal requirements: In many European countries and Australia, there is a statutory obligation to provide access for blind users when designing computer systems. In 1999 an Australian blind user successfully sued the Sydney Organising Committee for the Olympic Games under the Australian Disability Discrimination Act (DDA) due to his inability to order game tickets using Braille technology (Waddell 2002). Section 508 of the American Rehabilitation Act stipulates that all federal electronic information should be accessible to people with auditory, visual and mobility impairments.
- Exposure to more people: Disabled people and the elderly have good reason to use new technologies. People who are unable to drive or walk and those with mobility impairments can benefit from accessible online shopping. Communication technologies such as e-mail and mobile technology, can provide them with the social interaction they would otherwise not have.
- Better design and implementation: Incorporating accessibility into design results in an overall better system. Making systems accessible to the disabled will also enhance usability for users without disabilities.
- Cost savings: The initial cost of incorporating accessibility features into a design is high, but an accessible e-commerce site will result in more sales because more people will be able to access the site. Addressing accessibility issues will also reduce the legal expenses that could result from lawsuits by users who might want to enforce their right to equal treatment.

Guidelines to promote accessibility for users with disabilities were included in the US Rehabilitation Act (http://www.access-board.gov/508.html), an independent U.S. government under this Act a government agency devoted to accessibility for users with disabilities. The World Wide Web Consortium (W3C) adapted these guidelines (http://www.w3.org/TR/WCAG20/). According to Shneiderman et al (2014) the following accessibility guidelines were identified:

- Text alternatives. The idea behind a text alternative is to provide any non-text content so that it can be changed into other forms which users need, for example, into large print, Braille, speech, symbols or simpler language.
- Time-based media. If non-text content is time-based media, then text alternatives at least provide descriptive identification of the non-text content (e.g., movies or animations) and synchronise equivalent alternatives such as caption or auditory descriptions of the visual track with the presentation.
- Distinguishable. This guideline makes it easier for users to see and hear content and it

separates the foreground from the background. Colour is not used as the only visual means for conveying information, indicating an action, prompting a response or distinguishing a visual element.
- Predictable. It means the designer should make web pages appear and operate in predictable ways.

Advances in computer technology and the flexibility of computer software make it possible for designers to provide special services to users with disabilities. The flexibility of desktop, web and mobile devices makes it possible to design for people with special needs and disabilities. Below we consider two user groups with physical disabilities – visual and motor impairments – highlighting the limitations of normal input and output devices for them.

2.3.2.1 Users with Vision Impairments

Visually impaired people experience difficulties with output display besides the problems that the mouse and other input devices pose. Text-to-speech conversion can help blind users to receive electronic mail or read text files, and speech-recognition devices allow voice-controlled operation of some applications. Enlarging portions of a display or converting displays to Braille or voice output can be done with hardware and software that is easily obtainable. Speech generation and auditory interfaces are also used by sighted users under difficult conditions, for example, when driving an automobile, riding a bicycle or working in bright sunshine.

Reading and navigating text or objects on a computer screen is a very different experience for a user who cannot see properly. The introduction of graphical user interfaces (GUIs) was a setback for vision-impaired users, but technology innovations, such as screen readers facilitate the conversion of graphical information into non-visual modes. Screen readers are software applications that extract textual information from the computer's video memory and send it to a speech synthesiser that describes the elements of the display to the user (including icons, menus, punctuation and controls). Not being able to skim an entire page, the user has to navigate without any visual clues such as colour contrast, font or position (Phipps, Sutherland & Seale 2002). Pages that are split into columns, frames or boxes cannot be translated accurately by screen readers.

Using the mouse requires constant hand-eye coordination and reaction to visual feedback. This complicates matters for the visually impaired. They need to execute clicking and selecting functions by means of dedicated keys on a keyboard or through a special mouse that provides tactile feedback. Users with partial sight should be allowed to change the size, shape and colour of the onscreen mouse cursor, and auditory or tactile feedback of actions will be helpful.

With regard to keyboard use, visually impaired users require keys with large lettering, a high contrast between text and background, and even audible feedback when keys are pressed. Blind users usually access all commands and options from the keyboard; therefore, function and control keys need to be marked with Braille or tactile identification.

2.3.2.2 People with Motor Impairments

A significant proportion of the population have motor disabilities acquired at birth or through an accident or illness. Users with severe motor impairments are often excluded from using standard devices. Low-cost modifications can easily increase access without much effort. For those confined to bed, computers and the internet in particular provide a satisfying and stimulating means of interaction and give them access to resources, people and places to which they would not otherwise have access.

Users with physical impairments may have difficulties with grasping and moving a standard mouse. They also find fine motor coordination and selecting small on-screen targets demanding, if not impossible. Clicking, double clicking and drag-and-drop operations pose problems for these users. Designers must find ways to make this easier, for example, by letting the mouse vibrate if the cursor is over the target or implementing “gravity fields” around objects so that when the cursor comes into that field, it is drawn towards the target. Another solution is provided through trackballs that allow users to move the cursor using only the thumb. Severely physically impaired users may be able to move only their heads; therefore, either head-operated or eye tracking devices are required to control on-screen cursor movements or head-mounted optical mice. Speech input is another alternative, but there are still high error rates (especially if the user’s speech is also affected by the impairment) and it can only be used in a quiet environment.

Keyboards need to be detachable so that they can be positioned according to the user’s needs and there must be adequate grip between the keyboard and desktop so that the user cannot accidentally move the keyboard around. Individual keys should be separated by sufficient space and should not require much force to press. Oversized keyboards, key guards to guide fingers onto keys, and software-enabled sticky keys are possible solutions for users who experience uncertain touch. Some users prefer mouse sticks or hand splints to hit buttons. Designers can adapt the interface so that everything is controlled with a single button.

Users with hearing impairments use computers to convert tones to visual signals and communicate by e-mail in an office environment. Then there are telecommunication devices for the deaf (TDD or TYY) that enable telephone access to information, to train or airplane schedules, and to services (Shneiderman et al 2014). Improving designs for users with disabilities is an international concern.

Complete activity 2.5

2.4 Culture

Another perspective on individual differences has to do with cultural, ethnic, racial or linguistic backgrounds. It seems obvious that users who were raised learning to read Japanese or Chinese will scan a screen differently from users who were raised to read English or Afrikaans. Users from cultures that have a more reflective style or a great respect for ancestral traditions may prefer other interfaces than those chosen by users from cultures that are more action-oriented or novelty-based. Mobile device preferences may also vary across cultures and rapidly changing styles.

The term “culture” is often wrongly associated with national boundaries. Culture should rather be defined as the behaviour typical of a certain group or class of people. Culture is conceptualised as a system of meaning that underlies routine and behaviour in everyday working life. It includes race and ethnicity as well as other variables and is manifested in customary behaviours, assumptions and values, patterns of thinking and communicative style. According to Shneiderman et al (2014), designers are still struggling to establish guidelines for designing for multiple languages and cultures.

Nisbett (2003) compared the thought patterns of East Asians and Westerners and classified them as holistic and analytic respectively. Holistically-minded people tend to perceive a situation globally whereas analytically-minded people tend to perceive an object separately from the context and to assign objects to categories. Based on this distinction, Yong and Lee (2008) compared how these two groups view a web page. They found distinct differences. For example, holistically-minded people scan the whole page in a non-linear fashion, whereas analytically-minded people tend to employ a sequential reading pattern.

As software producers expand their markets by introducing their products in other countries, they face a host of new interface considerations. The influence of culture on computer use is constantly being researched, but there are two well-known approaches that designers follow when called on to create designs that span language or culture groups:

- *Internationalisation* refers to a single design that is appropriate for use worldwide among groups of nations. This is an important concept for designers of web-based applications that can be accessed from anywhere in the world by absolutely anybody.

- *Localisation*, on the other hand, involves the design of versions of a product for a specific group or community with one language and culture. The simplest problem here is the accurate translation of products into the target language. For example, all text (instructions, help, error messages, labels) might be stored in files so that versions in other languages could be generated with no or little programming. Hardware concerns include character sets, keyboards and special input devices. Other problems include sensitivity to cultural issues such as the use of images and colour.

User interface design concerns for internationalisation are numerous and internationalisation is full of pitfalls. Early designs were often forgiven for their cultural and linguistic slips, but the current highly competitive atmosphere means that more effective localisation will often produce a strong advantage. Simonite (2010) reports on the online translation services that now makes it possible to have web content immediately translated into other languages. These services use a technique called statistical machine translation that is based on statistical comparison of previously translated documents. This creates rules for future translation. In 2010 Google's translate services could translate between 52 different languages although the translations contained errors and needed some human intervention (Simonite 2010).

There are many factors that need to be addressed before a software package can be internationalised or localised. These can be categorised as overt and covert factors:

- *Overt factors* are tangible, straightforward and publicly observable. They include dates, calendars, weekends, day turnovers, time, telephone number and address formats, character sets, collating order sequence, reading and writing direction, punctuation, translation, units of measures and currency.

- *Covert factors* deal with the elements that are intangible and depend on culture or special knowledge. Symbols, colours, functionality, sound, metaphors and mental models are covert factors. Much of the literature on internationalising software has advised caution in addressing covert factors such as metaphors and graphics. This advice should be heeded to avoid misinterpretation of the meaning intended by the developers or inadvertent offence to the users of the target culture.

An example of misinterpretation is the use of the trash can icon in the Apple Macintosh user interface. People from Thailand do not recognise the American trash can because in Thailand trash cans are actually wicker baskets. Some visuals are recognisable in certain cultures, but they convey a totally different meaning. In the United States, the owl is a symbol of knowledge but in Central America, the owl is a symbol of witchcraft and black magic. A black cat is considered bad luck in the US but good luck in the UK. Similarly, certain colours hold different connotations in different cultures.

One culture may find certain covert elements inoffensive, but another may find the same elements offensive. In most English-speaking countries, images of the ring or OK hand gesture is understood correctly, but in France it means “zero”, “nothing” or “worthless”. In some Mediterranean countries, the gesture means that a man is homosexual. Covert factors will only work if the message intended in those covert factors is understood in the target culture. Before any software with covert factors is used, the software developers need to ensure that the correct information is communicated by validating these factors with the users in the target culture.

2.5 Personality and Gender

Some people dislike computers or get anxious when they have to use them; others are attracted to or eager to use any new kind of technology. Often members of these divergent groups disapprove or are suspicious of members of the other community. Even people who enjoy using computers may have different preferences regarding interaction styles, the pace of interaction, graphics versus tabular presentations, dense versus sparse data presentation, step-by-step work versus all-at-once work, and so on. These differences are important. A clear understanding of personality and cognitive styles can be helpful in designing systems for a specific community of users.

Despite fundamental differences between men and women, clear patterns of preferences in interaction have been documented. Social network sites such as Facebook and Twitter, tend to have more female subscribers. Huff and Cooper (1987) in their study on sex bias in educational software, found a bias when they asked teachers to design educational games for boys or girls. The designers created game-like challenges when they expected boys as their users, and more conversational dialogues when they expected girls as users. When told to design for students, the designers produced “boy-style” games.

It is often pointed out that the majority of video arcade game players and designers are young males. There are female players for any game, but popular choices among women for early video games were “Pacman” and its variants, plus a few other games such as “Donkey Kong” or “Tetris”. We can only speculate as to why women prefer these games. One female reviewer labelled Pacman as “oral aggressive” and could appreciate the female style of play. Other women have identified the compulsive cleaning up of every dot as an attraction. These games are distinguished by their less violent action and soundtrack. Also, the board is fully visible, characters have personality, softer colour patterns are used, and there is a sense of closure and completion. Can these informal conjectures be converted to measurable criteria and then validated? Can designers become more aware of the needs and desires of women, and create video games that will be more attractive to women than to men?

Turning from games to office automation, the predominant male designers may not realise the effect on female users when the command names require the users to KILL a file or ABORT a program. These and other potentially unfortunate mistakes and mismatches between the user interface and the user might be avoided by paying more attention to individual differences among users.

2.6 Age

Historically, computers and computer applications have been designed for use by adults for assisting them in their work. Consequently, in many accepted definitions of human-computer interaction and interaction design, there is a hidden assumption that users are adults. In definitions of HCI there are, for example, references to users' "everyday working lives" or the organisations they belong to. Nowadays, however, computer users span all ages. Applications are developed for toddlers aged two or three and special applications and mobile devices are designed for the elderly. User groups of different ages can have vastly different preferences with regard to interaction with computers.

The average age of the user population affects interface design. It is an indication of the level of expertise that may be assumed. In many instances, it affects the flexibility and tolerance of the user group. This does not always mean that younger users will be more flexible. They are likely to have used a wider range of systems and may have higher expectations. Age also determines the level of perceptual and cognitive resources to be expected from potential users. By this we mean that our ability to sense (perception) and process (cognition) information declines over time. Many user interfaces fail to take these factors into account.

Below we look at two special user groups – young children and the elderly – in detail.

2.6.1 Young Children

Child-computer interaction has emerged in recent years as a special research field in human-computer interaction. Children make up a substantial part of the larger user population. Whereas products for adult users usually aim to improve productivity and enhance performance, children's products are more likely to provide entertainment or engaging educational experiences. Applications designed for use by children in learning environments have completely different goals and contexts of use than applications for adults in a work environment (Inkpen 1997). While adults' main reasons for using technology are to improve productivity and to communicate, children do it for enjoyment. Another reason for distinguishing between adult and child products is young children's slower information processing skills that affect their motor skills and consequently their use of the mouse and other input devices (Hutchinson, Druin & Bederson, 2007).

Computer technology makes it possible for children to easily apply concepts in a variety of contexts (Roschelle et al 2000). It exposes them to activities and knowledge that would not be possible without computers. For example, a young child who cannot yet play a musical instrument can use software to compose music. People opposed to the use of computers by young children have warned against some potential dangers. These include keeping children from other essential activities, causing social isolation and reduced social skills and reducing creativity. There is general agreement that young children should not spend long hours in front of a computer, but computers do stimulate interaction rather than stifle it. Current advances in technology make it possible to create applications that offer highly stimulating environments and opportunities for physical interaction. New tangible and robotic interfaces are changing the way children play with computers (Plowman & Stephen 2003). The term "computer" in child-computer interaction refers not only to the ordinary desktop or notebook computer, but also to programmable toys, cellular phones, remote controls, programmable musical keyboards, robots and more. Tanaka, Cicourel and Movellan (2007, https://www.pnas.org/content/104/46/17954/tab-article-info) determined in their study that children treated the robot different than they treat each other (videos and clips are available on their site to view) (see figure 2.3).

Figure 2.3: Children interacting with the robot QRIO (https://www.pnas.org/content/104/46/17954 (Copyright (2007), National Academy of Sciences)

One way to address the concerns about the physical harm in spending too much time inactively in front of a computer screen is to develop technology that require children to move around. Dance mats that use sensory devices to detect movement are widely available. Computer vision and hearing technology can also be used to create games that use movement as input. A widely used commercial application that uses movement input is Sony's EyeToy™. The EyeToy is a motion recognition USB camera used with Sony's Play Station 2. It can detect movement of any part of the body, but most EyeToy games involve arm movements. An image of the player is projected on the screen to form part of the game space (see figure 2.4). Depending on the game context, certain areas of the screen are active during the game. Players must move so that their hands on the projected image interact with screen objects that are active in the game. For example, they have to hit or catch a moving ball. In other words, the user manipulates screen elements through his or her projected image.

Figure 2.4: Projected images of children playing Sony EyeToy games (Game Vortex, 2008) From http://www.psillustrated.com/psillustrated/soft_rev.php/2686/eyetoy-play-2-ps2.html

Clearly, technology has become an important element of the context in which today's children grow up and it is important to understand its impact on children and their development. According to Druin (1996), we should use this understanding to improve technology so that it supports children optimally. The development of any technology can only be successful if the designers truly understand the target user group. Knowledge of children's physical development and familiarity with the theories of children's cognitive development are thus essential when designing for them. The way children learn and play, the movies and television programmes they watch, and the way they make friends and communicate with others, are influenced by the presence of computer technology in their everyday lives. For this reason, Druin (1996) believes it is critical that designers of future technology observe and involve children in their work. When designing for children, the important thing is to accommodate them so that they can perform activities on the computer that are at their level of development.

Children uses focus on entertainment and education. Educational technology such as LeapFrog (http://www.leapfrog.com), designed educational packages for pre-readers using computer-controlled toys, music generators and art tools (Shneiderman et al 2014). As children's reading skills mature and they gain more keyboard skills, a wider range of desktop applications, web services and mobile devices can be incorporated. When children develop into teenagers, they can even assist parents and elderly users. This growth path identified by Shneiderman et al (2014) is followed by children who have access to technology and supportive parents as well as peers. But there are children who are not that privileged and lack the financial resources, supportive learning environment or access to technology. These constraints often frustrate them in their use of technology.

When designing for children, it is important to incorporate educational acceleration, facilitate socialisation with peers, and foster self-confidence that is normally associated with skill mastery. Shneiderman et al (2019) recommend that educational games should promote intrinsic motivation and constructive activities as goals. When designing for children, designers need to consider not only children's desire for challenge but also parents' requirements relating to safety. Children find it easy to deal with some level of frustration, but they also need to know that they can clear the screen, start over, and try again without penalties or limited penalties. They don't tolerate inappropriate humour and prefer familiar characters, exploratory environments, and the capacity to repeat. For example, children replay a game far more than adults do.

It is also important for designers to take note of children's limitations such as:

- evolving dexterity – meaning that mouse dragging, double-clicking, and small targets cannot always be used
- emerging literacy – meaning that written instructions and error messages are not effective
- low level of abstraction – meaning that complex sequences must be avoided unless the child uses the application under adult supervision
- short attention span and limited capacity to work with multiple concepts simultaneously (Shneiderman et al 2014).

According to Shneiderman et al (2014), the use of technology such as playful creativity in art and music, and writing combined with educational activities in science and math are areas which should inspire the development of children’s software. Educational materials can be made available to children at libraries, museums, government agencies, schools and commercial sources to enrich learning experiences. Educational material can also provide a basis for children to construct web resources, participate in collaborative efforts and contribute to community projects.

2.6.2 The Elderly

Owing to advances in health care technologies and living standards, the human life span is constantly increasing. This means that the population of older people is steadily growing, and that older people are more active than before. Although people now live longer, many of them will still develop some degenerative disabilities due to their advanced age (Darzentas & Miesenberger 2005).

The elderly have often been ignored as users of computers since they are assumed to be both dismissive of and unable to keep up with advancing technology. According to Shneiderman et al (2014), if designers understand human factors involved in aging, they can create user interfaces that facilitate access by older adult users. The stereotype that senior citizens are averse to the use of new technologies is not necessarily true (Dix, Finlay, Abowd & Beal, 2004). They do, however, experience impairments related to their vision, movement and memory capacity (Kaemba 2008) that affect the way they interact with devices. They have problems with mouse use because they complete movements slowly and have difficulty in performing fine motor actions such as cursor positioning. Moving the mouse cursor over small targets may be difficult for senior users, and double-clicking actions may be problematic, especially for users with hand tremors.

Shneiderman et al (2014) identified some benefits relating to senior citizens and their use of technology, for example, improved chances of productive employment and opportunities to use writing, e-mail and other computer tools. The benefits to society include seniors who share their valuable experience and offer emotional support to others. Senior citizens can also communicate with their children and grandchildren by e-mail or on social media. Many designers adapt their designs to cater for older adults because the world’s population ages and gets much older than in the past. According to Shneiderman et al (2014), desktop, web and mobile devices can be improved for all users by providing better control over font sizes, display contrast and audio levels. Hart et al (2008) recommend the following improvements of interfaces used by senior citizens: easier-to-use pointing devices, clean navigation paths as well as consistent layouts and a simpler command language.

The dexterity of our fingers decreases as we age, so elderly users may experience many difficulties typing long sequences of text on a keyboard. Keyboards that can easily be reached, have sufficient space between keys, provide audible or tactile feedback of pressed keys, and a high contrast between text and background may be required. Networking projects such as the San Francisco-based SeniorNet, provide elderly users over the age of 50 with access to and education about computing and the internet. The key focus of SeniorNet is to enhance elderly users' lives and to enable them to share their knowledge and wisdom (Shneiderman et al 2014). Nintendo's Wii also discovered that computer games are popular with elderly users because it stimulates social interaction, practises their sensory and motor skills such as eye-to-hand coordination, enhances their dexterity and improves their reaction time. In their study, Shneiderman et al (2014) also discovered that there was some fear of computers among elderly users and that they believed that they were incapable of using computers. But after a few positive experiences with computers, for example, sharing photos, exploring e-mail and using educational games, the fear gave way, and they were satisfied and eager to learn. Most of the mechanisms for supporting users with motor impairments described in section 2.3.2.2 are applicable to elderly users.

Many senior users find the text size on typical monitors too small and require more contrast between text and background. Even more so on small displays of mobile phones. Touch screens solve some of the interaction problems, but older users' habit of a finger along a text line while reading can result in unintended selections (Kaemba 2008). Clearly, the physical, social and mental contexts of the elderly differ from that of younger adults. The needs and preferences of adult technology users can therefore not be transferred to the elderly.

Complete activity 2.6

2.7 Expertise

The way in which a system is designed, built and sold depends on the intended users, on whether they are experts or novices. In the former case, designers must build upon existing skills. Issues such as consistency with previous interfaces, are absolutely critical. In the case of novice users, designers must provide a higher level of support. They must also anticipate some of the learning errors that can arise during interaction. It is difficult to begin the development process if designers are unaware of such general characteristics of their user population. Some people may only have partial information about how to complete a task. This is the typical situation of novice users of a computer application. They will need procedural information about what to do next. Experts, on the other hand, will have well-formed task models and do not need guidance. It follows, therefore, that novel task designers may have greater flexibility in the way that they implement their interface. In more established applications, expert users will have well-developed task structures and may not notice or adapt so quickly to any changes introduced in a system.

A number of models of skill levels have been developed to provide an explanation of how users operate at the different levels. The model in figure 2.5 shows the differences between users with different degrees of information about an interactive system. At the lowest level, the knowledge-based level, they may only be able to use general knowledge to help them understand the system. Designers can exploit this to support novice users. For example, in the Windows desktop, inexperienced users can apply their general knowledge in several ways, but sometimes with an unwanted effect. To recover a deleted file, a user might think he has to empty the recycle bin (waste bin). This is a dangerous approach. If they lack knowledge, then users are forced to guess.

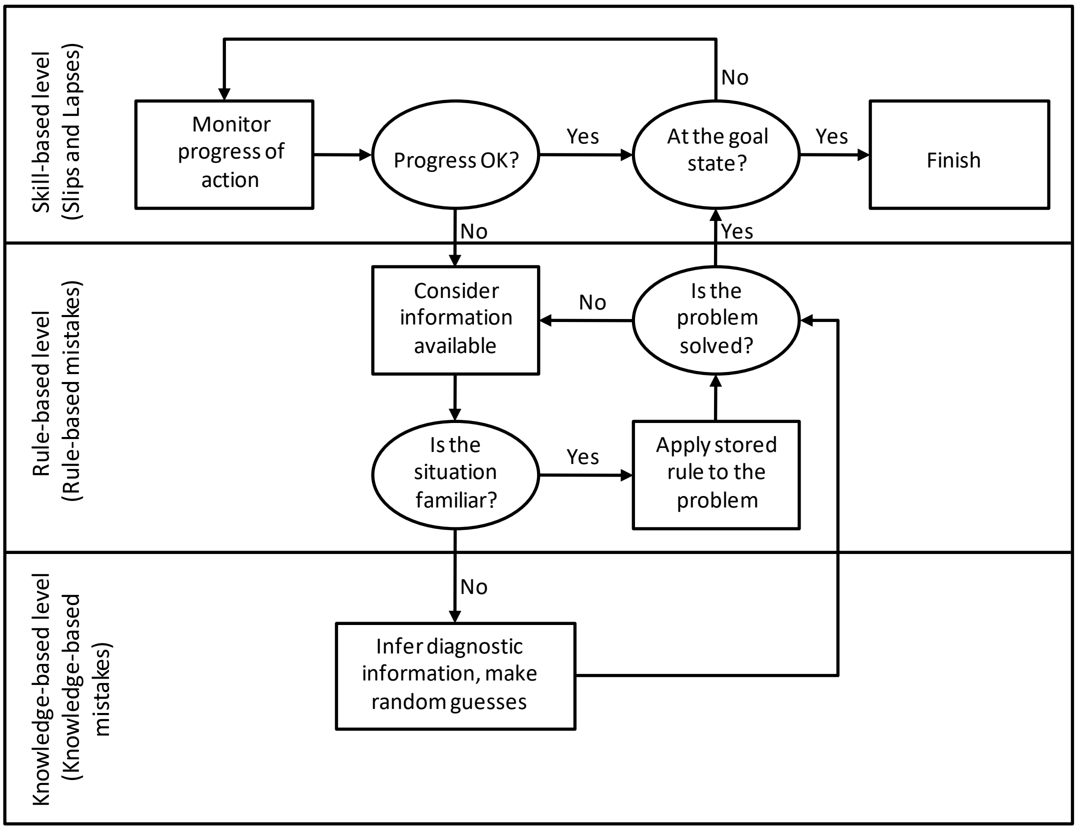


Figure 2.5: Three levels of expertise

The second level of interaction introduces the idea that users apply rules to guide their use of a system. This approach is slightly more informed than the use of general knowledge. For example, users will make inferences based on previous experience. This implies that designers should develop systems that are consistent. Similar operations should be performed in a similar manner. If this approach is adopted, then users can apply the rules learned with one system to help them operate another, for instance: “To print this page, I go to the File menu and select the option labelled Print”. There are two forms of consistency:

- *Internal consistency* refers to similar operations being performed in a similar manner within an application. This is easy to achieve if designers have control over the finished product.

- *External consistency* refers to similar operations being performed in a similar manner between several applications. This is hard to achieve as it involves the design of systems in which the designer may not be involved. This is the reason why companies such as Apple and IBM publish user interface guidelines.

Operating a user interface by referring to rules learned in other systems can be hard work. Users have to work out when they can apply their expertise. It also demands a high level of experience with computer applications. Over time, users will acquire the expertise that is required to operate a system. They will no longer need to think about previous experience with other systems and will become skilled in the use of the system. This typifies expert use of an application (the skill-based level in figure 2.5).

What designers should always keep in mind is that the more users have to think about using the interface, the less cognitive and perceptual resources they will have available for the main task.

2.8 The Errors People Make

2.8.1 Types of Error

People make errors routinely. It is part of human nature. There are several forms of human error or mistakes. Norman (1999) distinguishes the following main categories:

- Mistakes (also called "incorrect plans"): This category includes incorrect plans such as forming the wrong goal or performing the wrong action with relation to a specific goal. Situations in which operators adopt unsafe working practices are examples of this. These can arise either through a lack of training, poor management or through deliberate negligence. Mistakes are thus the result of a conscious but erroneous consideration of options.

- Slips: Slips are observable errors and result from automatic behaviour. They include confusions such as the confusion between left and right.

So, with a slip the person had the correct goal but performed the incorrect action; with a mistake the goal was incorrect.

The difference between mistakes and slips: humans will make slips so designers should design in such a way that makes the consequences of slip errors less irreversible. That is one of the reasons why emergency buttons are big and red. Mistakes occur when users don't know what to do because they haven't learned or haven't been taught to use something properly, for example, if someone uses an old Xbox game controller like a motion-sensitive Wiimote and waves it through the air instead of pressing the buttons. Slips occur mostly in skilled behaviour; when the user does not pay proper attention. Users who are still learning don't make slips (Norman 1999).

Norman (1999) distinguishes between the following kinds of slips:

- Capture errors: This occurs when an activity that you perform frequently is executed instead of the intended activity. For example, when I, on the day I have leave, drop my child at the pre-school and without thinking drive to work instead of driving home.

- Description errors: This occurs when, instead of the intended activity, you do something that has a lot in common with what you wanted to do. For example, instead of putting the ice-cream in the freezer, you put it in the fridge.

- Data-driven errors: These errors are triggered by some kind of sensory input. I once asked the babysitter to write her telephone number in my telephone directory. Instead of her own number, she copied the number of the entry just above her own. She was looking at that entry to see whether that person's name or surname was written first.

- Mode errors: These occur when a device has different modes of operation, and the same action has a different purpose in the different modes. For example, a watch can have a time-reading mode and a stopwatch mode. If the button that switches on a light in time-reading mode is also the button that resets the stopwatch, one may try to read the stopwatch in the dark by pressing the light button and thereby accidentally clearing the stopwatch.

- Associative activation errors: These are similar to description errors, but they are triggered by internal thoughts or associations instead of external data. For example, our secretary's name is Lynette, but she reminds me of someone else I know called Irene. I often call her

Irene.

- Loss-of-activation errors: These are errors due to forgetfulness. For example, you find yourself sitting with the phone in your hand, but you have forgotten who you wanted to call.

Complete activity 2.7

2.8.2 The Cause of Human Error

What is the true cause of human error? In the aftermath of many major accidents, it is typical to hear reports of an “operator error” as the primary cause of the failure. This term has little meaning unless it is supported by a careful analysis of the accident. For example, if an operator is forced to manage as best he/she can with a bug-ridden, unreliable system, is an accident then his/her fault or that of the person who implemented the program? If bugs are the result of poorly defined requirements or cost cutting during testing, are these failures then the fault of the programmer or the designer?

What appears to be an operator error is often the result of management failures. Even if systems are well designed and implemented, accidents can be caused because operators are poorly trained to use them. This raises practical problems because operators are frequently ill-equipped to respond to low-frequency but high-cost errors. How then can companies predict these events that, although they rarely occur, are sufficiently critical that users and operators should be trained in the procedures to solve them?

Further sources of error come from poor working environments. Again, a system may work well in a development environment, but the noise, heat, vibration or altitude of a user’s daily life may make the system unfit for its actual purpose.

2.8.3 How to prevent Human Error

There is no simple way to improve the operational safety of computer systems. Short-term improvements in operator training will not address the fundamental problems created by mistakes and slips. Reason (1990) argues that errors are latent within each one of us and, therefore, we should never hope to engineer out human error. This pessimistic analysis has been confirmed by experience. Even organisations with an exemplary training and recruitment system such as NASA, have suffered from the effects of human error.

There are, however, some obvious steps that can be taken to reduce both the frequency and the cost of human error. In terms of cost, it is possible to engineer decision support systems that provide users with guidance and help during the performance of critical operations. These systems may even implement cool-off periods during which users’ commands will not be effective until they have reviewed the criteria for a decision. These systems engineering solutions impose interlocks on control and limit the scope of human intervention. The consequences are obvious when such locks are placed in inappropriate areas of a system.

It is also possible to improve working practices. Most organisations see this as part of an ongoing training programme. In safety-critical applications there may be continuous and on-the-job competence monitoring, as well as formal examinations.

When designing systems, one should keep in mind the kinds of errors people make. For example, minimising different modes or making the different modes clearly visible, will avoid mode errors. Users may click on a delete button when they meant to click on the save button (maybe the delete button is located where, in a different application, the save button was placed). To prevent the user from incorrectly deleting something important, the interface should request confirmation before going through with a delete action.

Complete activity 2.8

2.9 Conclusion

Most computer-based systems are developed for use by people. Different systems are aimed at different kinds of people. It is crucial that whoever design these systems understand who they design for and how interacting with the system may affect these users.

In this lesson tool, we tried to give you a taste of the complexity of human cognition and human nature. Our goal was to help you realise that human problems and errors with technology are often a design failure, and that good design always considers human capabilities and weaknesses.

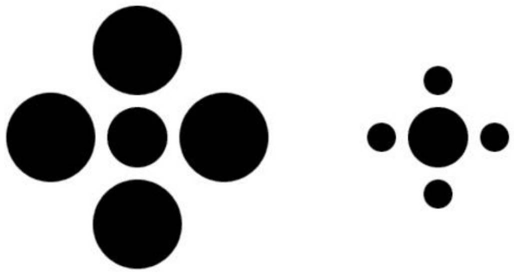
The final conclusion is that there is no such thing as the “average user”.

2.10 Activities

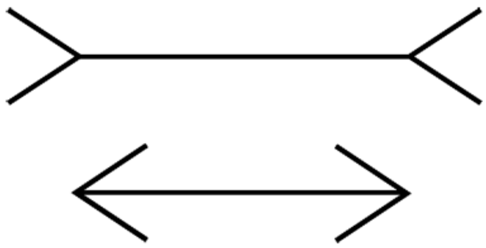
ACTIVITY 2.1

Look at the following images and answer the questions associated with each.

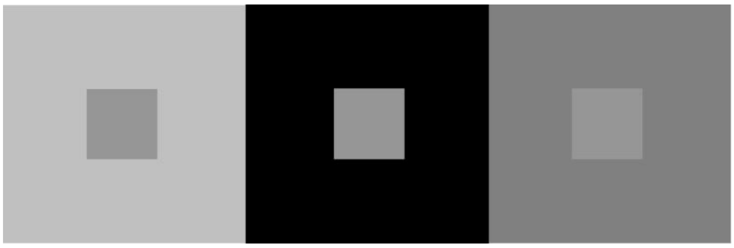
Which of the two circles in the middle are the biggest?



Which of the parallel lines are the longest?



Which of the squares in the middle is the darkest/lightest in colour?



If you used a ruler in 1 and 2, you would get the correct answer. 1 is called the Ebbinghaus illusion and 2 the Müller-Lyer illusion. In 3 there is actually no difference in the colour of the three inner squares. These examples show just how powerful external influences can be in our perception of things. You can also see that this is a good font to use if you want the reader to struggle to decipher it.

ACTIVITY 2.2

1. Draw up a table with two columns – one for STM and one for LTM – and list the differences between the two types of memory.
2. Give your own example of how the load on the user's STM can be relieved through thoughtful design of the interface.

ACTIVITY 2.3

Explain how we use cellular phones as knowledge in the world. Your answer should make it clear what is meant by the term "knowledge in the world".

ACTIVITY 2.4

Choose any computer-based activity you sometimes perform such as selecting and playing a song, writing and sending an e-mail, or submitting an assignment through myUnisa.

Name the activity. Now mention three broad categories of human resources we use in processing an action. Relate each category to how you would, in practice, use that resource in your chosen activity.

ACTIVITY 2.5

Stephen Hawking was a well-known physicist who has written influential books such as *A Brief History in Time*. Find information on him on the internet and then describe:

- the nature of his disability
- how it has affected his life
- how technology has helped him
- the mechanisms he used to interact with technology

ACTIVITY 2.6

Identify two cellphone users aged 15 or younger and another two aged 65 or older. Ask each of them to list three things they like about their cellphones and three things they do not like.

By comparing the lists, can you identify differences in the needs and preferences of users from the different age groups?

ACTIVITY 2.7

1. Give examples from your own experience or environment of each of the following: capture error, description error, data-driven error, mode error, associative activation error and loss-of-activation error.

2. Do research to find information on each of the following disasters. Can you identify the errors made as any of the types discussed above?
   - The Three Mile Island disaster
   - The Chernobyl disaster
   - Kegworth aircraft crash
   - Mars climate orbiter disaster

ACTIVITY 2.8

Using only information from this lesson tool of the study guide, identify 15 (fifteen) guidelines for the design of usable and/or accessible interfaces. Formulate them in your own words and in a way that will be useful to designers.

2.3 Lesson 3: Design Problems and Solutions

The content of this lesson is as follows:

CONTENTS

LESSON 3 – OUTCOMES

After studying this lesson, you should:

- understand the problems of designing an interactive system and specifically:
  - how the natural evolution of design is obstructed
  - common mistakes designers make
- understand and be able to apply the design practices and tools to avoid the problems with the design of interactive systems, namely:
  - mechanisms such as affordance, constraints, mapping, visibility and feedback
  - design guidelines, principles and standards

3.1 Introduction

Now that we have identified the key characteristics that distinguish different users, we will discuss some of the more concrete ways to make design decisions. We start by looking at some of the most common problems of interface design and then discuss how design problems can be overcome. We look at guidelines and principles for design compiled by some of the most influential researchers and authors in the field of HCI.

A substantial part of this lesson is based on the work of Don Norman presented in his most important book *The Design of Everyday Things*. The book was first published in 1989 and the 2000 edition hasn't changed much. Although the computer technology of 1989 was less sophisticated than that of today, the principles and advice given by Norman still apply. Reading the book will help you in this module and will change the way you look at everyday objects.

3.2 Design Problems

There are, potentially, many mistakes that designers can make. Norman (1999) points out the following as being the most problematic: the forces that work against evolutionary design, putting aesthetics first (that is, form over function), and designers regarding themselves as typical users.

3.2.1 Hampering the Natural Evolution of Design

Norman (1999) coined the term “evolutionary design”. It refers to the process whereby a product is gradually improved over time. Evolutionary design occurs when a design evolves through a cycle of testing, identifying problems, modification, redesign, retesting and remodification until a functional, aesthetically pleasing object is achieved. Good features are kept, and bad features are replaced with improved ones. This is a good process but, unfortunately, there are many obstacles in such a natural evolution. Three forces work against evolutionary design (Norman, 1999):

1. The demands of time

The new version of an object is often released even before the old one has been updated. Even if someone took the trouble to get feedback from users of the old version, there is not enough time to address the problems with the previous one. Microsoft often releases a new version of their operating system when there are still problems with it because releasing it on the promised date is more important than providing customers with a bug-free application. Hence the need for “service packs” and “hot fixes”.

2. Pressure to be distinctive

Each design must have features that distinguish it from previous versions so that consumers can be lured with statements such as “a new improved version”. Often the new model doesn’t even incorporate the good qualities of its predecessor.

3. The curse of individuality and market differentiation

Companies that manufacture the same type of product have to come up with a unique design which carries their signature. This means that if one company perfects a product, other companies that manufacture the same product often make an inferior product in the name of individuality. Of course, the quest for individuality can also lead to innovative solutions to real problems, but the goal should be to improve the product or solve the problem, not just to stand out.

Have you ever wondered why the keys on a computer keyboard are arranged in the order QWERTY? On the first ever rectangular typewriter keyboard, the keys were arranged alphabetically. This allowed for typing speeds that were too much for the typewriter’s mechanics – if the typing was too fast, the parts would get jammed. The solution was to rearrange the keys in a way that would slow down the typist. Here, a natural evolutionary process was followed, but the main driving force was the mechanical limitations of the instrument. People got so used to using the QWERTY keyboard, that it is still used today even if it was designed according to constraints that have disappeared long ago.

Complete activity 3.1

3.2.2 Common Design Mistakes

In lesson tool 2, we addressed the general mistake of designing for the typical user. Here we consider two further issues, namely overestimating the importance of aesthetics and designers regarding themselves as typical users. We will then briefly look at the problem of cluttered interfaces.

3.2.2.1 Putting Aesthetics above Usability

We cannot deny the fact that part of the appeal of Apple products is how they look. From the start, Apple Macintosh paid special attention to the aesthetics of their products. Aesthetics should, however, not take precedence over usability. Not long ago, computer applications could only be produced by computer scientists. Nowadays development tools allow people with limited or no programming knowledge to create applications such as web pages. The competitive commercial environment provides good motivation to employ graphic designers and artists to create attractive interfaces. Unfortunately, these designers do not always understand the importance of usefulness and usability.

An interface need not be an artwork to be aesthetically pleasing. One that is free of clutter, with the interface elements organised in a logical and well-balanced way, and that uses colour tastefully can provide visual pleasure to users who have to find their way through the interface.

Here again, the target user group should be considered. Young children prefer colourful interfaces with icons that move or twirl when the cursor moves over them, but this will annoy most older users. Culture may also determine what the user finds aesthetically pleasing.

Google.com is proof that a beautiful interface is not a prerequisite for a successful system. Google’s interface is pretty simple, but no one finds it offensive or unusable.

3.2.2.2 Thinking for the User

Designers sometimes believe that they know what the user would like, thinking that they can put themselves in the shoes of the user. Designers are expert users of technology, and they most often design applications that will be used by people who have far less knowledge of and exposure to technology. People tend to project their own feelings and beliefs onto others (e.g., mothers who force their children to wear jerseys because they themselves are cold). Designers are no different – they subconsciously build interfaces according to their own preferences and knowledge.

By the time a system is complete, the designers and developers know it so well that they will never be able to view it from the perspective of someone who encounters it for the first time. The user’s model of a system will be very different from that of a system designer. Users’ view of an application is heavily influenced by their tasks, goals and intentions. For instance, users may be concerned with letters, documents and printers. They are less concerned about the disk scheduling algorithms and device drivers that support their system.

Clearly, if designers continue to think in terms of engineering abstractions rather than the objects and operations in the users’ task, they are unlikely to produce successful interfaces. It is essential for designers to realise that they will make this mistake if they do not involve real users in the design process. The earlier in the process this happens, the better.

Another common error is to mistake the client for the end user and base the designs on the requirements specified by the client. For example, a university’s management may decide that they need a web-based learning management system that students can use to find information about their courses, download study material, and communicate with their lecturers and fellow students. They employ an IT development company to design, develop and implement the system according to their (the management’s) specifications. The designers should first determine who the end users will be (in this case the students) and test the specifications provided by university management against the requirements and preferences of these users.

3.2.2.3 Cluttering the Interface

Interfaces should provide users with enough information to allow them to perform their required task successfully. They should, however, avoid screen clutter. One reason is that it affects the aesthetics of the design, but even more important reasons are:

- It can be difficult for users to take in and understand the profusion of objects on the screen. Some may even be missed entirely.
- The more objects you present on the screen, the more meanings users will have to unravel.
- The more objects you present, the harder it is for users to find the ones that they really need.
- The more objects on the screen, the smaller the average size of each object. This makes it harder to select and manipulate individual screen components.

3.3 Design Solutions

In section 3.2, we highlighted some common design problems and gave some advice on how to avoid them. On a more positive note, this section will offer a range of mechanisms that support successful interface design. Principles for good interface design are not unique to interactive computer systems. Most basic principles for good interface design can be derived from Norman's (1999) good principles for the design of everyday things. Below we describe his well-known and widely accepted design concepts.

3.3.1 Affordance

The affordance of an object or interface refers to the perceived and actual properties that tell an observer or user how the object or interface can be used. For example, the handles of a pair of scissors usually have one hole that is smaller than the other – one is round and the other oval in shape. This tells the user how to hold the scissors. The thumb goes through the smaller hole and the other fingers go through the bigger hole. The stronger the clues provided by the affordances, the better the user will know what to do.

Consider the interface below:

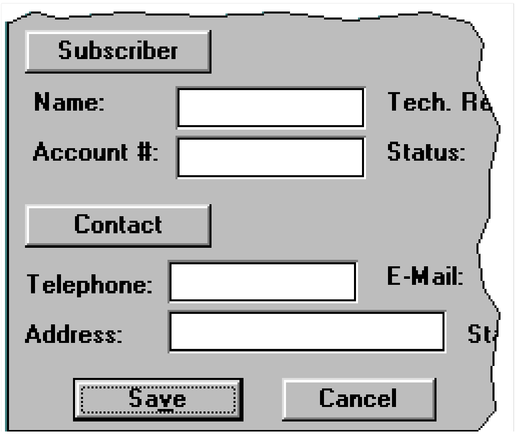


Figure 3.1 Example of affordance From http://hallofshame.gp.co.at/mdesign.htm

The Subscriber and Contact buttons in this interface fulfil the purpose of headings. Clicking on them will have no effect. But still, the user will think they should click on these 'buttons' since they invite clicking. So, we should use affordance to guide the user into taking the correct action, but we should also be careful to use controls for some purpose if they clearly afford another. Buttons should only be used for the purpose that they are designed for to invoke an action.

The Save and Cancel buttons on this interface also invite clicking, as they should.

3.3.2 Constraints

A constraint, in HCI terms, is a mechanism that restricts the allowed behaviour of a user when interacting with a computer system. For example, an ATM will only accept your card if you insert it into the slot the right way around. This is a physical constraint – it relies on properties of the physical world for its use. If a user cannot interpret the constraint easily, they will still have difficulty to perform the action. Often ATMs have a small icon next to the insertion slot to indicate to the user how the card should be inserted.

Not all constraints are physical. Constraints can also rely on the meaning of the situation (semantic constraint) or on accepted cultural conventions (cultural constraints). The fact that a red traffic light constrains a driver from crossing the road, is an example of a semantic constraint –drivers know that a red light means they should stop if they want to prevent an accident. They are not physically forced to stop, but the driver's interpretation of the situation makes him/her stop.

In some cultures, it is customary for a man to stand back to let a woman enter through a door first. Men who follow this custom are constrained by the cultural convention. An example of using a cultural constraint in interface design is to use a green button to go ahead with an operation or action and a red button to indicate the opposite. This follows the cultural convention that red means "stop" or "danger" whereas green means "go" or "OK".

Logical constraints refer to constraints that rely on the logical relationships between the functional and spatial aspects of a situation. Suppose there are two unmarked buttons on the doorbell panel of a house you visit. If you have no knowledge of the house or the people who live there, it will be difficult to decide which button to push. If you know that there is a flat to the left of the house, you can assume that the left-hand button is for the flat and the right-hand one for the house (assuming the occupants used some logic when installing the system). Natural mappings work according to logical constraints.

A forcing function is a type of physical constraint that requires one action before a next can take place (Norman 1999). The ATM example above is one of a forcing function. Another example is that you cannot switch on a front-loading washing machine unless the door is properly closed.

3.3.3 Mapping

Mapping refers to the relationship between two things, for example, the relationship between a device's controls and their movements, and the results of the actual use of these controls. A good mapping is one that enables users to determine the relationships between possible actions and their respective results. Programming your television's channels so that you get SABC1 by pressing 1 on the remote control, SABC2 by pressing 2, SABC3 by pressing 3 and e-TV by pressing 4 is an example of a natural mapping. In a computer interface, there should be good mapping between the text on buttons or menus and the functions activated by choosing those buttons or menu items. A standard convention in Windows interfaces is to use *Save* on a menu item that overwrites the current copy of a document and *Save As* when you want to create a new instance of the document. By now most people are familiar with this, but it would have been a better mapping to name the menu item as *Save a Copy*.

Natural mappings use physical analogy and cultural standards to support interpretation. Figure 3.2 shows some icons from a children's game. The *Page Backward* and *Page Forward* icons provide a natural mapping with their functions. They clearly depict a page, and the arrows indicate the direction of paging through the document. Their spatial orientation further strengthens the mapping – the left-hand one for backwards and the right-hand one for forward.

The traffic light icon, which is for exiting the page, does not. There is no logical, spatial or semantic connection between a traffic light and the exit operation.

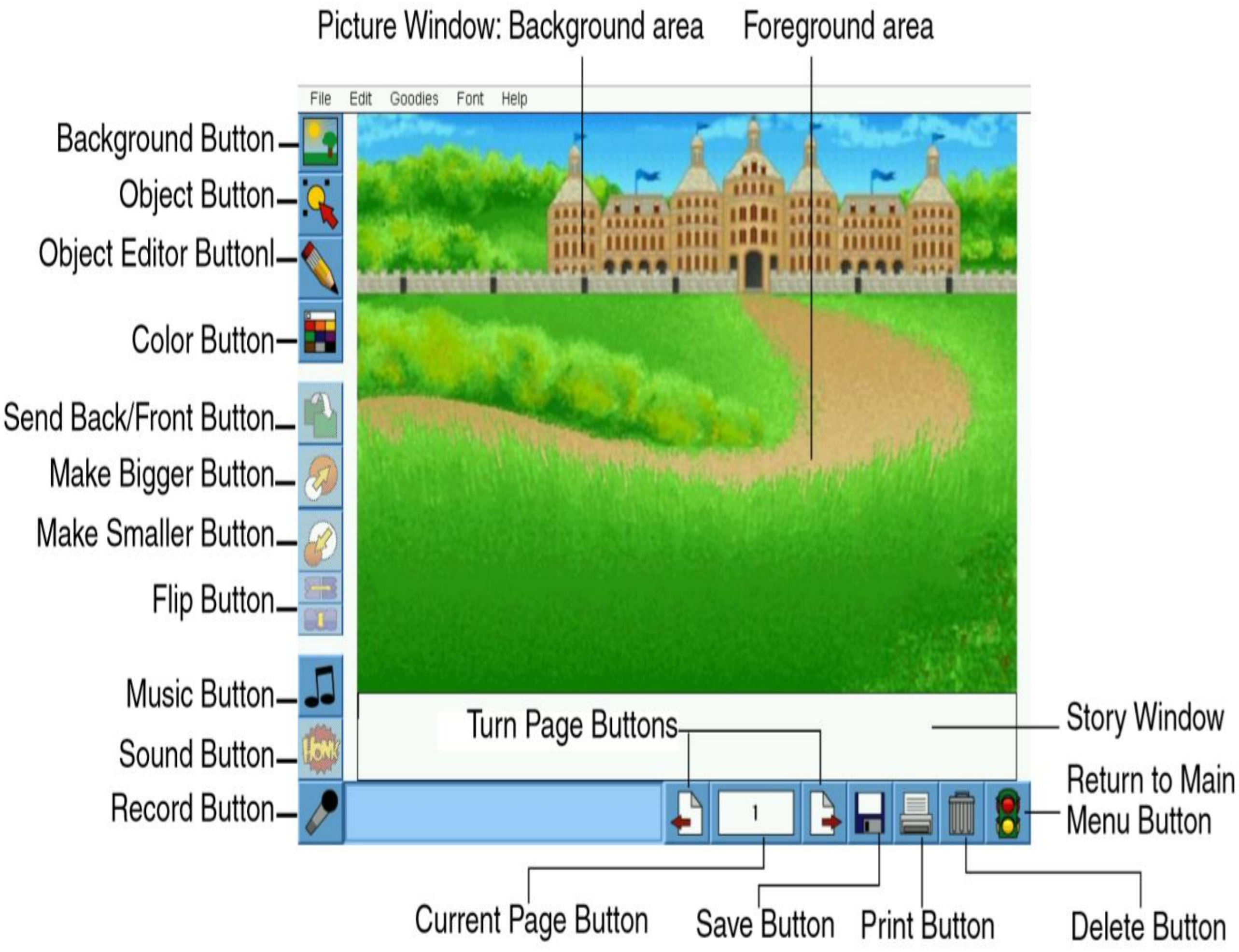


Figure 3.2: Icons from a children's game
http://gpisdtechhelp.pbworks.com/f/StorybookWeaverDeluxeUserGuide.pdf

3.3.4 Visibility

The parts of a system that are essential for its use must be visible. The visible structure of well-designed objects gives the user clues about how to operate them. These clues take the form of affordances, constraints and mappings. Visible signs (like letters or the colour) on salt and pepper shakers tell us which one is which. The main menu of Storybook Weaver Deluxe 2004 is given in figure 3.3 (the explanatory text provided in this figure does not appear on the interface). The absence of text labels to the icons makes it difficult for users to interpret them – especially young children who will not associate a light bulb with story ideas or a quill and inkpot with creating a new story. There are, in fact, text labels associated with the icons, but they only become visible if the mouse pointer is moved across the icons. However, there is no way for users to know this. This interface fails badly in terms of visibility.

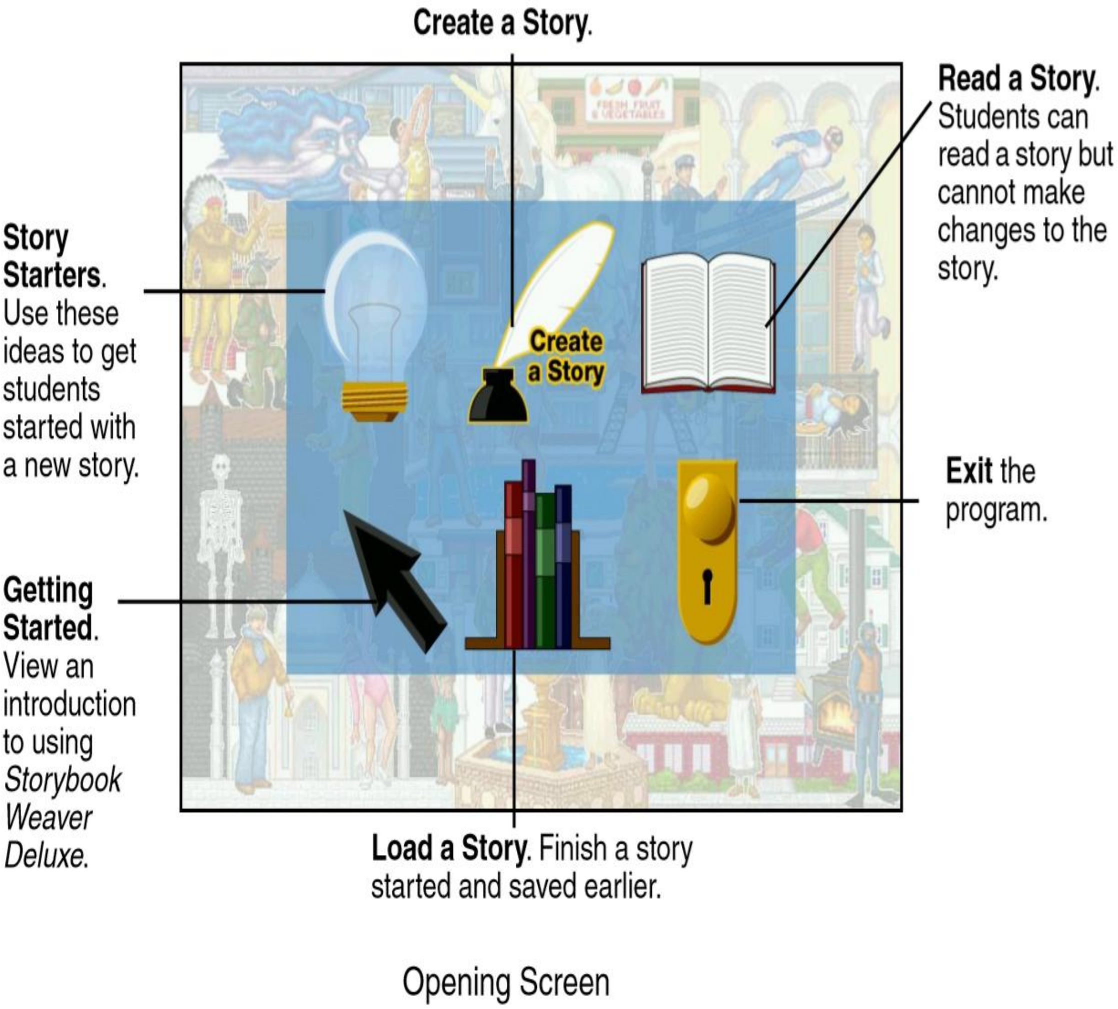


Figure 3.3: Opening screen of Storybook Weaver Deluxe 2004
http://gpisdtechhelp.pbworks.com/f/StorybookWeaverDeluxeUserGuide.pdf

Sound can also be used to make interface elements more visible. Often an error message has a sound attached to it to draw the user's attention to the problem. In products for children who cannot yet read, audio cues can be attached to icons instead of text labels. Sound calls our attention to an interface when there is new information, for example, a beep on a cellphone signals the arrival of a new message.

3.3.5 Feedback

Feedback is information that is sent back to the user about what action has actually been performed, and what the result of that action is. When we type, we know that we have pressed the keys hard enough if the letters appear on screen. Operations that take time are often indicated by a progress bar or a message stating that the process is under way. Without constant feedback, the interaction process will be very unsatisfactory.

Novices want more informative feedback to confirm their actions; frequent users want less distracting feedback.

Consider figure 3.4. It shows a window that appears directly after a user of Storybook Weaver Deluxe 2004 has clicked on the *Save As Web Document* option on the *File* menu. The web document is automatically saved in the user's *My Documents* folder in a subfolder called "Storybook Weaver Deluxe". The title of the story is used as the name of the web document. Is this suitable and adequate feedback for a young child?

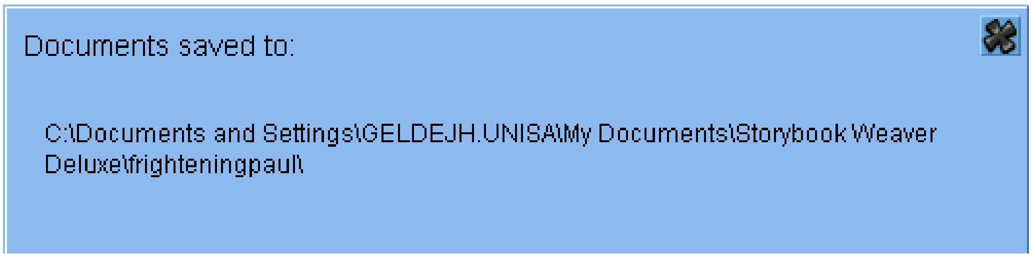


Figure 3.4 Feedback after saving a story as a web document
http://gpisdtechhelp.pbworks.com/f/StorybookWeaverDeluxeUserGuide.pdf

Complete activity 3.2

Sound is an important feedback mechanism. We know that the washing machine's door is closed properly when we hear the click. Sound feedback is extensively used in cars – my car beeps annoyingly at a steadily increasing volume when I drive without fastening my seatbelt and it also beeps when the petrol tank is close to empty. The absence of sound can also be a form of feedback. If you switch on a kettle and do not hear the water heating up after a while, it is an indication that something is wrong with the kettle.

Complete activity 3.3

3.3.6 Guidelines, Principles and Standards

The aim with design guidelines, standards and design principles is to help designers to improve the usability of their products by giving them rules according to which they can make design decisions (Dix et al 2004). Such rules restrict the range of design options and prevent the designer from making choices which are likely to harm the usability of the product. Dix et al classify design rules as standards or guidelines. Standards are usually set by national or international bodies and are authoritative and limited in application. Guidelines, on the other hand, are more general in application.

There are two types of design guidelines: low-level detailed rules and high-level directing principles. High-level principles are relatively abstract and apply to different systems whereas detailed rules are instructions that are application specific and do not need much interpretation.

The difference between design principles and usability principles is that design principles usually inform the design of a system, whereas usability principles are mostly used as the basis for evaluating prototypes and complete systems (Preece et al 2007; Preece et al 2019). Usability principles can be more prescriptive than design principles. In practice, some design or usability principles are referred to as "heuristics" (Preece et al 2007; Preece et al 2019).

Below we discuss some of the most prominent sets of guidelines, namely those of Dix et al (2014), Preece et al (2007), Preece et al (2019) and Shneiderman (1998).

3.3.6.1 Dix, Finlay, Abowd and Beale

Dix et al (2014) provide interface designers with a comprehensive set of high-level directing principles with the aim of improving the usability of interactive systems. They divide their principles into three categories, namely Learnability principles, Flexibility principles and Robustness principles. They summarise their principles in three tables that we reproduce below as tables 3.1, 3.3 and 3.5. In these tables our added explanations appear in italics. The related principles are described in tables 3.2, 3.4 and 3.6 respectively.

Learnability

Learnability refers to the ease with which users can enter a new system and reach a maximal level of performance. Dix et al (2014) identified five principles that affect the learnability of a computer-based system. They are defined in table 3.1.

Table 3.1: Principles that affect Learnability (from Dix et al (2004)**,** p 261)

Principle	**Definition**	**Related principles (explained in table 3.2)**
Predictability	Support for the user to determine the effect of future action based on past interaction history.	Operation visibility
Synthesisability	Support for the user to assess the effect of past operations on the current state. *To be able to predict future behaviour, a user should know the effect of previous actions on the system. Changes to the internal state of the system must be visible to users so that they can associate it with the operation that caused it.*	Immediate/Eventual Honesty
Familiarity	The extent to which a user's knowledge and experience in other real-world or computer-based domains can be applied when interacting with a new system. *The user's first impression is important here. Familiarity can be achieved through metaphors and the effective use of affordances that exist for interface objects. Clickable objects must look clickable, for example.*	Guessability, affordance
Generalisability	Support for the user to extend knowledge of a specific interaction within and across applications to other similar situations.	
Consistency	Likeness in input-output behaviour arising from similar situations or similar task objectives.	

We expand on the related principles mentioned above in table 3.2.

Table 3.2: Principles that relate to Learnability principles

Principle	Explanation
Operation visibility	The way in which the availability of possible next operations is shown to the user and how the user is informed that certain operations are not available.
Honesty	The ability of the user interface to provide an observable and informative account of any change an operation makes to the internal state of the system. It is immediate when the notification requires no further interaction by the user. It is eventual when the user has to issue explicit directives to make the changes observable.
Guessability and affordance	The way the appearance of the object stimulates a familiarity with its behaviour or function.

Flexibility

Flexibility refers to the many ways in which interaction between the user and the system can take place. The main principles of flexibility formulated by Dix et al (2004) are explained in table 3.3 and other principles that relate to these are described in table 3.4.

Table 3.3: Principles that affect Flexibility (from Dix et al (2004)**,** p 266)

Principle	Definition	Related principles
Dialogue initiative	Allowing the user freedom from artificial constraints on the input dialogue imposed by the system.	System/User pre-emptiveness
Multi-threading	Ability of the system to support user interaction pertaining to more than one task at a time.	Concurrent vs interleaving modality
Task migratability	The ability to pass control for the execution of a given task so that it becomes either internalised by the user or the system or is shared between them. *For example, a spellchecker does some of the work, but should ultimately let the user decide which words to replace.*	
Substitutivity	Allowing equivalent values of input and output to be arbitrarily substituted for each other.	Representation multiplicity, equal opportunity
Customisability	Modifiability of the user interface by the user or the system.	Adaptivity, adaptability

Table 3.4: Principles that relate to Flexibility principles

Principle	Explanation
System pre-emptiveness	This occurs when the system initiates all dialogue, and the user simply responds to requests for information. It hinders flexibility but may be necessary in multi-user systems where users should not be allowed to perform actions simultaneously.
User pre-emptiveness	This gives the user freedom to initiate any action towards the system. It promotes flexibility, but too much freedom may cause the user to lose track of uncompleted tasks.
Concurrent/ interleaved multi-threading	Concurrent multi-threading allows simultaneous communication of information pertaining to separate tasks. Interleaved multi-threading permits a temporal overlap between separate tasks, but at any time the dialogue is restricted to a single task.
Multimodality	Separate modalities (channels of communication) are combined to form a single input or output expression.
Representation multiplicity	Flexibility for the rendering of state information, for example, in different formats or modes.
Equal opportunity	Blurs the distinction between input and output at the interface – the user has the choice between what is input and what is output; in addition, output can be reused as input.
Adaptability	Refers to user-initiated modification to adjust the form of input and output. Users may, for example, choose between different languages or complexity levels.
Adaptivity	Refers to a system-initiated modification to customise the user interface automatically. Here the system should observe the users' behaviour (for example, repeated attempts at tasks) and determine their level of expertise in order to adjust the complexity level of tasks.

Robustness

Robustness refers to the level of support that users is given for the successful achievement and assessment of their goals. Table 3.5 summarises Dix et al's (2004) robustness principles and table 3.6 lists the supporting principles.

Table 3.5: Principles that affect Robustness (from Dix et al (2004), p 270)

Principle	Definition	Related principles
Observability	Ability of the user to evaluate the internal state of the system from its perceivable representation. The user compares the current state with his or her intention within the task-action plan.	Browsability, static/dynamic defaults, reachability, persistence, operation visibility
Recoverability	Ability of the user to take corrective action once an error has been recognised.	Reachability, forward/backward recovery, commensurate effort
Responsiveness	How the user perceives the rate of communication with the system. Response time is the amount of time needed by a system to inform the user of state changes. When this is not instantaneous, the system should give some indication that the task is in progress.	Stability
Task conformance	The degree to which the system services support all the tasks that the user wishes to perform and in the way the user understands them.	Task completeness, task adequacy.

Table 3.6: Principles that relate to Robustness principles

Principle	Explanation
Browsability	This allows the user to explore the current internal state of the system via the limited view provided at the interface. The user should be able to browse to some extent to get a clear picture of what is going on, but negative side effects should be avoided.
Static/Dynamic defaults	Static defaults are defined within the system or acquired at initialisation. Dynamic defaults evolve during the interactive session (for example, the system may pick up a certain user’s input preference and provide this as the default input where applicable).
Reachability	The possibility of navigation through the observable system states.

Persistence	Deals with the duration of the effect of a communication act and the ability of the user to make use of that effect. Audio communication persists only in the user's memory whereas visual communication remains available for as long as the user can see the display.
Backward recovery	Involves an attempt to undo the effects of previous interaction in order to return to a prior state.
Forward recovery	Involves the acceptance of the current state and negotiation from that state towards the desired state.
Commensurate effort	If it is difficult to undo a given effect on the state, then it should have been difficult to do in the first place.
Stability	The invariance in response times for identical or similar computational resources.
Task completeness	Refers to the coverage of all the tasks of interest and whether or not they are supported in a way the user prefers.
Task adequacy	This addresses the user's understanding of the tasks.

3.3.6.2 Preece, Rogers and Sharp

Preece et al (2019) discuss two types of design goals in interaction design, namely usability goals and user experience goals. The usability goals focus on aspects such as effectiveness and learnability, whereas the user experience goals are concerned with the quality of the user's experience with the system and focus on aspects such as aesthetics and enjoyment.

Usability Goals

Preece et al (2019) have identified six usability goals that will ensure that users' interaction with technology is effective and enjoyable. We summarise these goals in table 3.7.

Table 3.7: Preece et al's (2019) usability goals

Usability goal	Explanation
Effectiveness	A general goal that refers to how well a system does that for which it was designed.
Efficiency	This has to do with how well a system supports users in carrying out their work. The focus is on productivity.
Safety	Protecting the user from dangerous conditions and undesirable situations.

Utility	The extent to which a system provides the required functionality for the tasks it was intended to support. Users should be able to carry out all the tasks in the way they want to do them.
Learnability	How easily users learn to use the system.
Memorability	How easy it is to remember how to perform tasks that have been done before.

User Experience Goals

According to Preece et al (2019), how the user feels about a product, irrespective of its efficiency, effectiveness, learnability and so on, plays an important role in it being well accepted or not. To provide users with positive experiences of interaction, designers should attend to features that will make the product satisfying, enjoyable, engaging, pleasurable, exciting, entertaining, helpful, motivating, aesthetically pleasing, supportive of creativity, cognitively stimulating, rewarding, fun, provocative, surprising, emotionally fulfilling, challenging and sociability enhancing (Preece et al 2019). Features that make a product boring, frustrating, annoying or overly cute should be avoided.

Clearly one would not spend too much design effort on making a spreadsheet application entertaining or emotionally fulfilling, but these user experience goals are applicable to many new technologies in different application areas. Factors that may support the fulfilment of these user experience goals include attention, pace, interactivity, engagement and style of narrative (Preece et al 2019).

Design Principles

According to Preece et al (2019), design principles are prescriptive suggestions that help designers to explain or improve their designs. Instead of telling the designer exactly how to design an interface, they inspire careful design by telling the designer what will work and what not. They discuss a number of design principles that we summarise in table 3.8. You will see that they correspond with Norman's principles as discussed in section 3.2.

Table 3.8: Summary of Preece et al's (2019) discussion of design principles

Principle	**Explanation**
Visibility	The more visible the available functions are, the better users will be able to perform their next task.
Feedback	This involves providing information (audio, tactile, verbal or visual) about what action the user has performed and what the effect of that action was.
Constraints	These restrict the actions a user can take at a specific point during the interaction. This is an effective error prevention mechanism.

Mapping	This has to do with the relationships between interface elements and their effect on the system. For example, clicking on a left-pointing arrow at the top left-hand corner of the screen takes the user to the previous page and a right-pointing arrow in the right-hand corner takes the user to the next page.
Consistency	This is similar to consistency as defined by Dix et al (2004).
Affordance	This refers to an attribute of an object that tells users how it should be used. In an interface, it is the perceived affordance of an interface element that helps the user to see what it can be used for. Whereas a real button affords pushing, an interface button affords clicking. A real door affords opening and closing, but an image of a door on an interface affords clicking in order to open it.

3.3.6.3 Shneiderman

Shneiderman (1998) and Shneiderman et al’s (2014) principles for user-centred design are divided into three groups, namely recognition of diversity, golden rules and prevention of errors.

Recognise Diversity

Before the task of designing a system can begin, information must be gathered about the intended users, their tasks, the environment of use and the frequency of use. According to Shneiderman, this involves the characterisation of three aspects relating to the intended system: usage profiles, task profiles and interactions styles. We explain these in table 3.9.

Table 3.9: Three aspects relating to recognition of diversity (Shneiderman 1998; Shneiderman et al 2014)

Aspect	**Explanation**
Usage profiles	Designers must understand the intended users. Shneiderman lists several characteristics that should be described. Those that apply to young children are age, gender, physical abilities, level of education, cultural or ethnic background and personality. Designers should find out whether or not all users are novices, if they have experience with the particular kind of system, or if a mixture of novice and expert users is expected. Different levels of expertise will require a layered approach whereby novices are given a few options to choose from and are closely protected from making mistakes. As their confidence grows, they can move to more advanced levels. Users who enter the system with knowledge of the tasks should be able to progress faster through the levels.
Task profiles	A complete task analysis should be executed, and all task objects and actions should be identified. Tasks can also be categorised according to frequencies: frequent actions, less frequent actions and infrequent actions. Frequent actions include special keys such as arrow keys, insert and delete; less frequent actions comprise *Ctrl* or pull-down menus; and infrequent actions include changing the printer format.
Interaction styles	Suitable interaction styles should be identified from those available. Shneiderman mentions menu selection, form fill-in, command language, natural language and direct manipulation. In lesson tool 4, we discuss most of the currently available interaction styles.

The Eight Golden Rules for Interface Design

Shneiderman et al (2014) suggest eight principles of design that are applicable to most interactive systems. They overlap to some extent with those of Dix et al (2004) and Preece et al (2019), and are mostly self-explanatory. They are:

1. Strive for consistency.
2. Cater to universal usability, for example, enable frequent users to use shortcuts.
3. Offer informative feedback.
4. Design dialogues to yield closure (the completion of a group of actions).
5. Offer error prevention and simple error handling.
6. Permit easy reversal of actions.
7. Support an internal locus of control (the user should feel in control of the system and not vice versa).
8. Reduce short-term memory load.

Prevent Errors

The last group of principles proposed by Shneiderman (1998) and Shneiderman et al (2014) pertain to designing to prevent the user from making errors. Errors are made by even the most experienced users, for example, the users of cellphones, e-mail, spreadsheets, air-traffic control systems and other interactive systems (Shneiderman et al 2014). One way to reduce a loss in productivity due to errors is to improve the error messages provided by the computer system. A more effective approach is to prevent the errors from occurring. The first step towards attaining this goal is to understand the nature of errors (we discussed this in lesson tool 2). The next step is to organise screens and menus functionally by designing commands and menu choices that are distinctive and by making it difficult for users to perform irreversible actions. Shneiderman et al (2014) suggest three techniques which can reduce errors by ensuring complete and correct actions:

- Correct matching pairs: For example, when a user types a left parenthesis, the system displays a message somewhere on the screen that the right parenthesis is outstanding. The message disappears when the user types the right parenthesis.
- Complete sequences: For example, logging onto a network requires the user to perform a sequence of actions. When the user does this for the first time, the system can store the information and henceforth allow the user to trigger the sequence with a single action. The user is then not required to memorise the complete sequence.
- Correct commands: To help users to type commands correctly, a system can, for example, employ command completion which will display complete alternatives as soon as the user has typed the first few letters of a command.

3.3.6.4 Design Standards

Standards concern prescribed ways of discussing, presenting or doing something. The aim is to achieve consistency across products of the same type. We are familiar with standards in many aspects of life: there are standard colours for electrical wiring and standard shoe and garment sizes. Standards have been set for many areas of the computer industry: paper sizes, transmission protocols, compilers, character set representations and so forth. For the last decade or so there has been a move to introduce standards in interface design.

Standards for interactive system design are usually set by national or international bodies to ensure compliance with a set of design rules by a large community. Standards can apply to either the hardware or the software used to build the interactive system.

Standardisation in interface design offers various benefits. It:

- provides a common terminology so that designers know that they are discussing the same concept
- facilitates program maintenance and allows for additional facilities to be added
- gives similar systems the same look and feel so that elements are easily recognisable
- reduces training needs because knowledge can be transferred between standardised systems
- promotes the health and safety of users who will be less likely to experience stress or surprise due to unexpected system behaviour

On the other hand, a user interface design rule that is rigidly applied without taking the target user's skills, psychological and physical characteristics or preferences into account, may reduce a product's usability. Standards must therefore always be used together with more general interface design principles such as those proposed by Dix et al (2004), Preece et al (2019) and Shneiderman et al (2014) as discussed above.

3.4 Conclusion

In this lesson tool, we looked at some of the things designers of system interfaces do wrong, but we focused mostly on how to design correctly. In doing this, we gave an overview of some of the most prominent sets of guidelines and principles for interface design.

It is important to realise that design guidelines do not provide recipes for designing successful systems. They only provide guidance and do not guarantee optimum usability. Even armed with very good guidelines, a designer should still make an effort to understand the technology and the tasks involved, the relevant psychological characteristics of the intended users, and what usability means in the context of the particular product.

Guidelines can help designers to identify good and bad options for an interface. They also restrict the range of techniques that can be used while still conforming to a particular style, but they can be very difficult to apply. In many ways they are only as good as the person who uses them. This is a critical point because many companies view guidelines as a panacea. The way to improve an interface is not just to draft a set of rules about how many menu items to use or what colours make good backgrounds. We cannot emphasise enough that users' tasks and basic psychological characteristics must be considered. Unless you understand these factors, guidelines will be applied incorrectly.

3.5 Activities

ACTIVITY 3.1
Discuss any example of a design that reflects evolutionary design in the good sense, or that demonstrates how the forces against evolutionary design have prevented a product from naturally evolving into something better.

ACTIVITY 3.2
Suggest ways in which the feedback in figure 3.4 can be made more suitable for a six- or seven-year-old user.

ACTIVITY 3.3

After a tiring journey, you check into a guest house. In your room, there is a welcome tea tray and kettle. But, oh dear, the kettle cord dangles and there's no sign of an electric wall plug near the kettle. After filling the kettle in the bathroom, you go down on your hands and knees and find an extension cord coming from behind a cupboard. Relieved, you plug in the kettle, but the kettle has an up-down switch, and the extension has an embedded left-right switch. Neither switch indicates ON or OFF and there are no friendly little red lights to show that current is flowing. By trial and error, you work out the right combination. The tea, eventually, was really good.

Referring to Norman's principles of design (affordance, mapping, constraints, visibility and feedback), clearly identify which of these concepts are relevant to the given case, define them and then show whether they were implemented or not.

2.4 Lesson 4: Interaction Design

CONTENTS

LESSON 4 – OUTCOMES

After studying this lesson, you should:

- know and be able to describe the different interface types that current computer systems have
- understand and be able to give examples of specific interaction design techniques, namely:
  - low-fidelity prototyping
  - high-fidelity prototyping
  - conceptual design including interface metaphors

4.1 Introduction

In this lesson, we discuss a small selection of topics that are associated with interaction design. INF3720 – the third-level module on HCI – covers the topic of interaction design in detail. Here we look at the following:

- the different forms that interfaces of interactive systems can take
- some specific interaction design techniques
- an overview of the evaluation of interactive systems

4.2 Interface Types

According to Preece et al (2019), it is important to, already at the conceptual design phase, consider the interface type to be used. It is done for both design and practical purposes. Preece et al (2007, 2014, 2019) give an overview of the different types of interfaces. We provide a brief description of a few of these.

4.2.1 Command-based

In early interfaces, command-based interface types were most commonly used. The user had to respond to a prompt symbol appearing on the computer display to which the system responded. Another characteristic of command-based interfaces is pressing a combination of keys (e.g., Alt+Control+Delete). Some commands also form part of the keyboard such as Enter, Delete and Undo, together with function keys (F5 to display a Power Point presentation). The command line interface has been superseded by graphic interfaces that incorporate commands such as menus, icons, keyboard shortcuts or pop-ups.

An advantage of command line interfaces is that users find them easier and faster to use than equivalent menu-based systems when performing certain operations as part of a complex software package, for example, CAD environments to enable expert designers to interact rapidly and precisely with the software. Command line interfaces have also been developed for impaired people to enable them to interact in virtual worlds.

4.2.2 Advanced Graphical Interfaces

The term “graphical user interface” (GUI) refers to any interactive system that uses pictures or images to communicate information. This is an extremely wide definition. It includes keyboard-based systems that only use graphics to present data. It also comprises walk-up and use systems where users interact by selecting portions of a graphical image. The challenge for software designers today is to design GUIs that are best suited for tablet, smartphone and smartwatch interfaces. Instead of just using a mouse and keyboard as input, the default for most users is to swipe and touch using a single finger when they want to scroll down, navigate through applications, browse and interact with digital content (Preece et al 2019).

The strength of GUIs is the way they support interaction in terms of (Shneiderman 1998, Shneiderman et al 2014):

- Visibility: Graphical displays can be used to represent complex relationships in data sets that otherwise would not have been apparent. This use of graphics is illustrated by the bar charts and graphs that were introduced in the section on requirements elicitation.
- Cross-cultural communication: It is important that designers exploit the greatest common denominator when developing interaction techniques. Text-based interfaces have severe limitations in the world market. Graphical interaction techniques are less limited. In particular, ISO standards for common icons provide an international graphics language (a lingua franca).
- Impact and animation: Graphical images have a greater intuitive appeal than text-based interfaces, especially if they are animated. The use of such techniques may be beneficial in terms of the quality and quantity of information conveyed. It may also be beneficial in improving user reactions to the system itself.

Some weaknesses of GUIs are:

- Clutter: There is a tendency to clutter graphical displays with a vast array of symbols and colours. This creates perceptual problems and makes it difficult for users to extract the necessary information. Graphical images should be used with care and discretion if people are to understand the meanings associated with the symbols and pictures.
- Ambiguity: Graphical user interfaces depend on the fact that users are able to associate

some semantic information with the image. In other words, they have to know the meaning of the image. As we said in our earlier discussion, they have to interpret the mappings. Users' ability to do this is affected by their expertise and the context in which they find the icon.

- Imprecision: There are some contexts in which graphical user interfaces simply cannot convey enough information without textual annotation. For instance, using the picture of an expanding and shrinking bar to represent the changing speed of a car is probably not a good idea when there is an important difference between 120 and 121 kilometres per hour.
- Slow speed: Graphical presentation techniques are unsuitable if there are relatively low bandwidth communication facilities or low-quality presentation devices. The performance problems need not relate directly to the graphical processing. Network delays may delay the presentation of results; this violates the rapid criteria for direct manipulation.
- Difficulty finding specific windows: When too many windows are open, it can be difficult for users to find a specific window.

Advanced graphical interfaces involve interactive animations, multimedia, virtual environments and visualisations. Multimedia includes graphics, text, video, sound and animations with which the user can interact. It supports quick access to multiple representations of information and is well suited for training, education and entertainment. The problem with multimedia is that users tend to favour animations and video clips and easily ignore accompanying text and static diagrams.

Virtual reality and virtual environments are graphical simulations that create the illusion that the user is part of the environment. It gives users the experience of operating in 3D environments in ways that are not possible in the real world. Virtual objects can be very true to life. Users in a virtual environment have either a first-person perspective where they see the environment through their own eyes, or a third-person perspective where they see the environment through the eyes of an avatar, an artificial representation of a real person (Dix et al 2004).

4.2.2 Web-based Interfaces

Web-based interfaces are graphical interfaces that are located on servers connected to the internet and are accessed by users through web browsers. Web design is restricted by the available bandwidth and the associated download time. Although, in first-world countries, high bandwidth is available to most people, large numbers of internet users in developing countries do not have fast internet access. Nowadays websites can have most of the characteristics of advanced graphical interfaces, but uncluttered design and easy accessibility of the required information are still preferable to web pages filled with flashing advertisements and a great deal of graphics and animations. Users should always know where they are, what they can find there, and where they can go to next. Web design still relies heavily on the use of text.

The main advantage of web-based interaction is that it provides users with access to large volumes of information at the click of a button. Sophisticated search engines such as Google, makes it easy to search for information on specific topics. Unfortunately, there are also large amounts of irrelevant information to search through and, since practically anybody can load information onto the web, much of it is not trustworthy.

Another important advantage of web-based interaction is the social aspect. It allows people to connect quite easily with anybody anywhere in the world.

4.2.3 Speech Interfaces

A speech interface allows the user to talk to a system that has the capacity to interpret spoken language. It is commonly used in systems that provide specific information (e.g., flight times) or perform a specific transaction (e.g., buy a movie ticket). Technology such as speech-enabled screen readers and speech-operated home control systems (e.g., for switching appliances on and off), can be useful to people with disabilities. Speech recognition also include word processors, page scanners and web. Current technology allows for far more natural-sounding speech than early synthesised speech. Speech interfaces in applications for children who cannot yet read will expand the possibilities that technology can offer them. Speech interface has come of age and is more advanced than in the early 1990s. Corporations that utilise telephones use speech interfaces a whole lot more. Speech-to-text systems have also become more popular, for example, Dragon Dictate (Preece et al 2019). One of the most popular speech technology applications is called routing. Companies use an automated speech system to enable users to reach one their services with the use of caller-led speech. Speech-based apps enable people to use them with their mobile devices, which makes it more convenient than any text-based entry. Users can articulate their queries by using Google Voice or Apple Siri. Mobile translators also use speech technology applications.

But there are some disadvantages to these kinds of interfaces:

- They are relatively difficult to develop.
- They may not be adaptable to different accents, voice pitches and speech defects (e.g., lisping).
- They may misinterpret what the user says.
- The voice response may sound unnatural.

Complete activity 4.1

4.2.4 Pen, Gesture and Touchscreen Interfaces

Personal digital assistants (PDAs) come with a pen for making on-screen selections, writing notes or making freehand sketches. Objects can also be manipulated by making swiping or stroking gestures. Pen-based interfaces are also suitable for large displays. Through a process called “digital ink” that uses sophisticated handwriting recognition and conversion techniques, text written on a PDA screen or tablet PC can be converted into digital text. Gesture-based input involves camera capture and computer vision to detect people’s arm and hand gestures. This makes sign language interpreting systems possible. The latest systems use sensor technologies to detect touch, bend and speed of movement. Touchscreens allow users to manipulate screen objects with their fingers. Two hands can, for example, be used to stretch an object in two different directions at the same time. Pen-based interfaces have mainly been designed to interact with tablets and large displays instead of mouse or keyboard input and selecting items.

These kinds of interfaces can increase the speed and accuracy of input, and users use natural gestures to interact. They also provide options for users who may have difficulty using the mouse and keyboard. Disadvantages are that the flow of interaction may be interrupted, incorrect options may accidentally be chosen, and movement and handwriting may be misinterpreted. Another disadvantage of using pen-based interactions on small screens such as a PDA, is that handwritten notes can also be converted and saved as standard typeface text using a pen-based device. Another benefit of a digital pen is that it allows the user to annotate existing documents easily and quickly, for example, spreadsheets, presentations and diagrams. Children can write with a stylus on a tablet PC without making too many spelling errors.

Touchscreens at walk-in kiosks, ticket machines at the movies, museum guides, airports, ATMs and tills or in cars and GPS systems have been around for quite a while (Preece et al 2019). They function by detecting the presence and location of human touch and react on finger tapping, swiping, flicking, pinching and pushing. Touchscreens work especially well when zooming in and out of maps or moving objects such as photos or scrolling through lists, for example, a music selection on a car's touchscreen display or a phone list.

4.2.5 Mobile Interfaces

These are interfaces designed for handheld devices such as cellphones that are intended for use on the move. The space limitations compel designers to use buttons for multiple purposes. For example, a single key on a cellphone can represent up to five characters and each is associated with a predefined number of presses of the key. Elderly users find this particularly hard to use. A small screen size restricts the font size and the amount of information that can be displayed on the screen.

A number of cellphones have been designed specifically for elderly users. They have larger buttons with larger text on the keys and a larger display that allows for the use of bigger fonts. Unfortunately, many older users do not buy their own phones but receive them as gifts from children and grandchildren whose own contracts allow for an upgrade of the phone model.

Handheld devices such as smartphones and iPads, differ from PCs and laptops in terms of their size, portability and interaction style. Early cellphones were equipped with hard-wired, small and physical keyboards. Letters had to be pressed. Most modern smartphones are touch based with virtual pop-up keyboards and are interacted with by finger tapping. Tablets and smartphones are also increasingly used in classrooms (Preece et al 2019). Smartphones can also be used to download contextual information by scanning barcodes in the physical world.

4.2.6 Multimodal Interfaces

In multimodal interfaces, different ways of interacting – including touch, sight, sound and speech – are combined so that users can experience or control information in various ways. Different input or output methods are used simultaneously, for example, speech and gesture or eye-gaze and gesture. Speech and touch combinations are already being used, but few other multimodal interfaces are commercially available.

They allow more flexible interaction and can support users with disabilities or very young users. There are several disadvantages to multimodal interfaces: input needs to be calibrated for accurate interpretation; they are complex and difficult to implement; and they are still very expensive.

4.2.7 Shareable Interfaces

These interfaces allow more than one user to interact with the system by providing multiple (sometimes simultaneous) inputs. Tabletop environments that detect touch input from multiple users at the same time already exist. They use an array of embedded antennae that each transmits a unique signal. The users each sit on their own chair or mat which has a receiver installed. Through the user's body, a signal goes from the tabletop to the receiver and tells the computer which antenna was touched.

Shareable interfaces provide a large interactional space and supports flexible group work and sharing of information, which enable group members to create content jointly and at the same time. One example of a shareable interface is Roomware. It is designed to integrate interactive furniture pieces, walls, table and chairs. Users can work as a network and position items.

Disadvantages are that separating personal and shared workspaces requires specialised hardware and software and correct positioning at the interface. These interfaces are also expensive to develop.

4.2.8 Tangible Interfaces

Hornecker and Buur (2006:437) describe tangible interaction as encompassing “a broad range of systems and interfaces relying on embodied interaction, tangible manipulation and physical representation (of data), embeddedness in real space and digitally augmenting physical spaces”.

These interfaces use sensor-based interaction. Physical objects that contain sensors react to user input which can be in the form of speech, touch or the manipulation of the object. The effect can take place in the physical object (e.g., a toy that reacts to a child’s spoken commands) or in some other place (e.g., on a computer screen). Technology incorporating tangible interfaces have been increasingly used since 2005. Sensors are usually RFID tags which can be stickers, cards or disks that store and retrieve data through a wireless connection with an RFID transceiver (Preece et al 2019). Tangible interfaces have been used for urban planning and storytelling technologies, and are generally good for learning, design and collaboration. Physical representations of real-life, manipulatable objects enable the visualisation of complex plans. Physical objects and digital representations can be positioned, combined and explored in dynamic and creative ways.

Tangible interfaces are particularly suitable for young children. Children’s body movements and their ability to touch, feel and manipulate things are important for developing sensory awareness and, therefore, also for general cognitive development (Antle 2007). Tangibles can also help children to understand abstract concepts because these are often based on their comprehension of spatial concepts and how they use their bodies in space (Antle 2007).

PETS (Personal Electronic Teller of Stories) (Montemayor, Druin & Hendler 2000) is a tangible storytelling system that allows children to create their own interface – a soft, robotic toy. Figure 4.1 shows an example of an interface built by children. These toys provide the interface between the child and the storytelling software that is located on a computer.

Figure 4.1: An example of a PETS creature (University of Maryland HCI Lab, 2008) From: http:// www.cs.umd.edu/hcil/pubs/screenshots/PETS

Preece et al (2019) identified several benefits of using tangible interfaces compared with other interfaces such as pen-based GUI. For example, physical objects and digital representations can be positioned, combined and explored in creative ways by enabling dynamic information to be presented in different ways. Physical objects can be held in both hands and combined and manipulated in ways not possible with other interfaces. More than one person can explore the interface together and objects can be placed on top of, beside and inside each other. These different configurations encourage different ways of representing and exploring a problems space.

Some of the problems with tangible interfaces are its development cost, inaccurate mapping between actions and their effects, and the incorrect placement of digital feedback.

4.2.9 Augmented- and Mixed-Reality Interfaces

In an augmented-reality interface, virtual representations are superimposed on physical devices and objects, whereas in a mixed-reality environment, views of the real world are combined with views of a virtual environment. Mixed-reality systems have been used for medical applications, for example, a scanned image of organs or an unborn baby is projected onto the body of the patient to help doctors to see what goes on inside someone's body. Another industry where augmented-reality interfaces are used is aviation. They show the real plane landing, taking off and taxiing. Aircraft pilots also use them in poor weather conditions (Preece et al 2019).

These interfaces enhance perception of the real world and support training and education (for example, in flight simulators). Everyday graphical representations such as maps, can be overlaid with additional dynamic information such as the ones used in fish finders, where maps are loaded and structures as well as fish are showed in detail. The fisherman can also map locations on the surface while a projector augments the maps with the projected information. To reveal digital information, the user opens the AR app on a smartphone or tablet and the content appears superimposed on what is viewed through the screen. AR apps have also been developed to guide a person holding a phone while walking. Real-estate apps combine an image of the residential property with its price per square metre (Preece et al 2019). However, the added information could become distracting, and users may have difficulty to distinguish between the real and the virtual world. These systems are also quite expensive.

4.2.10 Wearable Interfaces

These interfaces involve input and output devices that are integrated with normal apparel such as headgear or goggles. They are mobile and less restrictive than desk-based technologies or even mobile technologies. They create a sense of realism and provide immediate feedback which can be helpful in the detection of medical conditions. Some problems are still experienced with these interfaces: they are uncomfortable because of their size and weight, and their battery life is restricted.

A well-known example of a wearable interface is the sensor built into running shoes so that runners or their family members can monitor their progress during a race.

4.2.11 Multimedia

A multimedia interface implies the use of a combination of different media within a single interface, namely graphics, text, video, sound and animation. Multimedia interfaces also link the different combinations of media with other forms of interactivity. For example, users can click on a hotspot or link in an image or text in one screen that leads to another part of the programme where the user might find a video or animation. After viewing it, users can return to where they were or move to another place.

Multimedia interfaces are designed to be used with games that encourage users to explore different parts of a game or story by clicking on different parts of the screen. The assumption is that combining media and interactivity provides better ways of presenting information and of doing it in different formats. Hands-on interactive simulations have also been incorporated as part of multimedia learning environments. Multimedia interfaces have been developed for mostly training, educational and entertainment purposes.

One of the advantages of multimedia interfaces is its ability to facilitate rapid access to multiple representations of information. Multimedia interfaces work especially well in digital libraries because they provide an assortment of audio and visual materials on a certain topic. For example, if you want to know more about the liver, a typical multimedia-base encyclopaedia will provide:

- a video clip on liver functioning and the possibility of a liver transplant
- audio recordings of a physician talking about the causes of liver diseases
- static diagrams and animations of the liver
- hypertext describing the structure of the liver and its functioning

4.2.12 Virtual Reality

According to Preece et al (2019), virtual reality (VR) uses computer-generated graphical simulations to create the illusion of participation in a synthetic environment. VR refers to the experience of interacting with an artificial environment. Early VR was developed using head-mounted displays, but users found it difficult to use and uncomfortable to wear. Besides, it caused motion sickness and was too expensive. VR technology has advanced considerably since the 1990s. VR headsets are now more affordable and more accurate head tracking allows developers to create compelling games, movies and virtual environments.

VR is widely used in the retail and property sectors. For example, if you want to paint a room, you take a photo of it and then use an application designed to “paint” the walls in the colour you selected from a palette. This way, you see how it will look without physically painting the room. Similarly, prospective buyers can “walk” through a house that is for sale without physically doing so.

One advantage of VR interfaces is that it can provide opportunities for new kinds of immersive experience by enabling users to interact with objects and navigation in 3D space. 3D software toolkits make the programming of a virtual environment easier. For example, the toolkit Alice (www.alice.org) allows users to use mice, keyboards or joysticks as input devices.

One advantage of VR is the high fidelity of simulations of the real world compared to other forms of graphic interfaces such as multimedia. The illusion afforded by technology can indeed make virtual objects exceptionally life-like.

4.2.13 Robotic and Drone Interfaces

These are interfaces that enable users to move and steer a remote robot. Robots are computational devices that have the physical appearance and behaviour of humans or animals. They can be built to go into places too small or dangerous for humans or to do manual repetitive tasks. Domestic robots can be manipulated to help in the house or to assist the disabled and the elderly to pick up objects from the floor, to cook meals, and so on. Pet-like robots have been developed to host events or act as a companion. They contain embedded sensors that detect user behaviours and then respond to them.

Sony's Aibo (figure 4.2) is a robotic dog that can perform playful behaviour, wag its tail, walk, lie down and stand up, sit and shake hands (Bartlett, Estivill-Castro & Seymon 2004). Despite its robotic appearance, these features are enough to convince a child that it is a living being with feelings even if their attention is drawn to its robotic features.

Figure 4.2: AIBO (Hughes, 2001) From: http://the-gadgeteer.com/review/sony_aibo_review [1]

New Scientist magazine of 24 April 2010 reported that NASA intended to send a humanoid robot into space. The plan was to send a robot as part of the crew to perform mundane mechanical tasks. The robot called Robonaut (see figure 4.3) went on its first trip into space in September 2010, and researchers examined the influence of cosmic radiation and electromagnetic interference on its performance. It consisted of a head and torso with highly functional arms to manipulate tools.

Figure 4.3: Robonaut (From http://www.tgdaily.com/space-features/48312-nasa-builds-new-generation-robot-astronaut)

Videos of "Robot Pets of the Future" can also be viewed at http://youtu.be/wBFws1lhuv0.[1]

Drones are unmanned aircraft that are controlled remotely (Preece et al 2019). Initially, robots were used by hobbyists and later on by the military. Since then, drones have become more

[1] Although every effort has been made to trace the copyright holders, this has not always been possible. Should any infringement have occurred, the publisher apologises and undertakes to amend the omission in the event of a reprint.

affordable, accessible and easier to fly. They are employed in the entertainment industry to carry drinks and food around at festivals, in agriculture to fly over farmlands and vineyards to collect crop data, over grazing paddocks to find cattle, and in the wildlife industry to swoop over game parks to spot rhinos or rhino poachers. Compared to other forms of data collection devices, drones can fly very low and stream photos to a ground station where images are stitched onto maps and then used to determine, for example, the health of a crop and when would be the best time to harvest.

4.2.14 Information Visualisation and Dashboard Interfaces

Information visualisation (also called *infoviz*) are computer-generated graphics of complex data which are interactive and dynamic. Preece et al (2019) identified some aims of information visualisation interfaces. They are to amplify human cognition by enabling users to see patterns, trends and anomalies in the visualisation and to gain insight from it. Another objective is to facilitate discovery, decision-making and the explanation of phenomena.

Most information visualisation interfaces are used by experts and enable them to make sense of vast amounts of dynamic or everchanging information. For example, satellite images or research findings will take much longer to process if only text-based information is utilised. Shneiderman et al (2014) used tree maps to visualise file systems thereby enabling users to understand why they are running out of disk space, to realise how much space different applications take up, and also to view large image repositories that contain terabytes of satellite images. In the financial world, visualisation is utilised to represent changes in the prices of stocks and shares over time, and rollovers show additional information.

Dashboards have also become a popular form of visualising information. Dashboards show screenshots of data updated over periods of time that is intended to be read at a glance. Dashboards tend to be more interactive, and slices of data depict the current state of a system or process.

A disadvantage of dashboards identified by Preece et al (2019) is the poor visual design of dashboards by software vendors. Dashboards should provide digestible and legible information so that users can home in on what is important to them.

4.2.15 Consumer Electronics and Appliances

Consumer electronic appliances include machines for everyday use in public, at home or in a car, for example, washing machines, microwaves, dish washers, DVD players, vending machines, remotes and navigation systems. Consumer electronic appliances also include personal devices such as MP3 players, digital clocks and digital cameras. One characteristic that the everyday use of consumer electronic appliances have in common with personal devices is that users try to get something done in the shortest period of time, such as switching on the washing machine, watching a TV programme, buying a ticket, setting the time, or taking a snapshot. Consumers are unlikely to read through a manual to see how they are supposed to use the appliance.

Complete activity 4.2

4.3 Interaction Design Techniques

According to Preece et al (2019), the iterative interaction design process involves, firstly, to identify users’ needs and requirements; secondly, to develop alternative designs according to those requirements and then to build interactive versions (prototypes) of those designs and, finally, to evaluate the users’ experience of the product. Preece et al (2019) also concluded that good interaction design is the philosophy of user-centred design, which means that users are involved throughout the development.

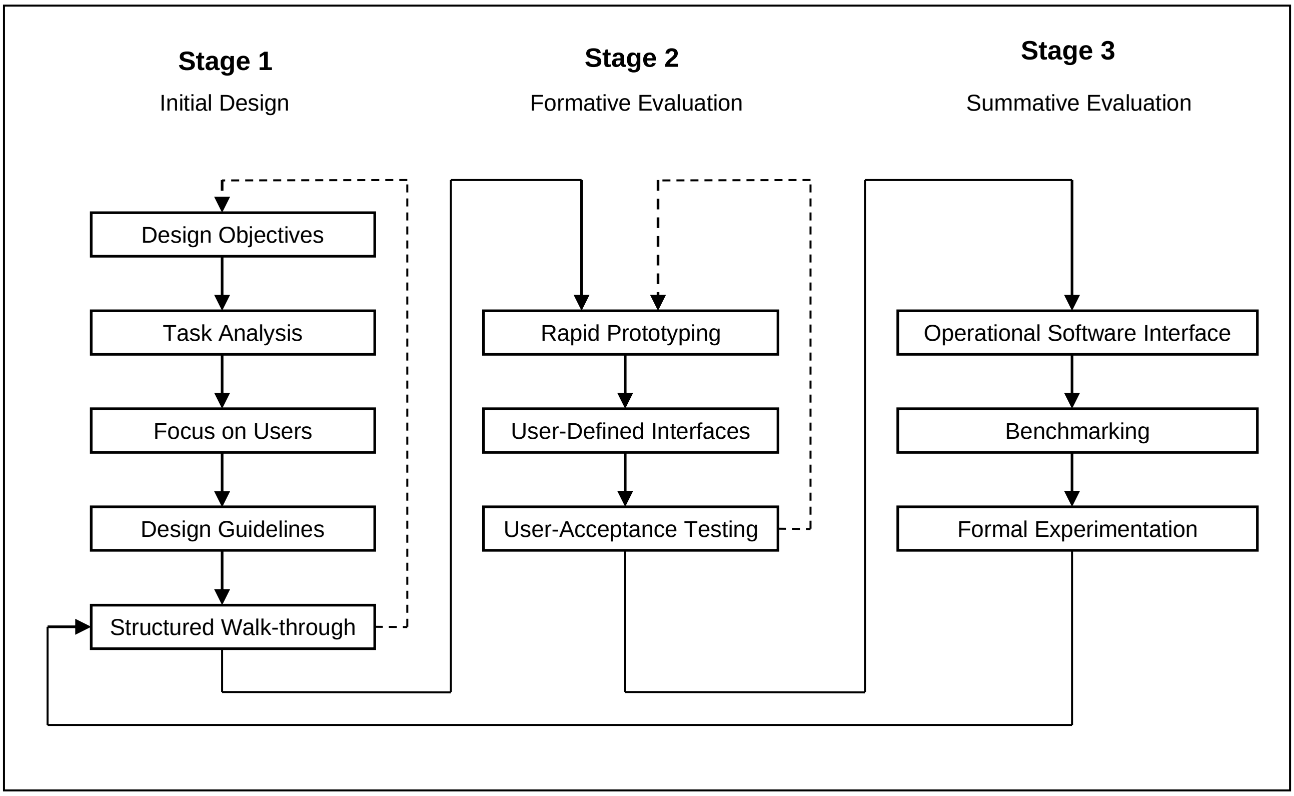


Figure 4.4: HCI Life Cycle (Williges & Williges, 1984)

More than 20 years ago, Williges and Williges (1984) produced their classic model of software development whereby interface design drives the overall design process. A graphical representation of their model appears in figure 4.4. Their idea is that by identifying user requirements early on in the software development process, code generation and modification effort will be reduced. It is not any different from what Preece et al (2019) advocate.

In lesson tools 2 and 3, we looked at some of the elements of stage 1 of this model, namely "Focus on users" and "Design Guidelines". In the remainder of this lesson tool, we discuss some of the components of stages 2 and 3. Stage 2 (formative evaluation) involves prototyping and an evaluation of the prototype, and stage 3 (summative evaluation) involves the evaluation of the completed system.

4.3.1 Prototypes

4.3.1.1 Definition and Purpose

Preece et al (2019:530) define a prototype as "a limited representation of a design that allows users to interact with it and to explore its usability". It can range from a simple, paper-based storyboard of the interface screens to a computer-based, functionally reduced version of the actual system. Prototyping has advanced into 3D printing technology. Companies use it to create and design a model from a software package and then print the prototype of, for example, soft toys or chocolates (Preece et al 2019).

Prototypes have several functions:

1. They provide a way to test different design ideas, especially during the evaluation of ideas.
2. They act as a communication medium within the design team – members test their ideas on the prototype and the team discusses their ideas.
3. They act as a communication medium between designers and users or clients. Using one or more prototypes, designers can explain to users and clients their own understanding of what the system should look like and what it should be able to do. The users can then respond to this by explaining how the prototype does or does not address their needs.
4. They help designers to choose between alternative designs.

4.3.1.2 Low-Fidelity Prototypes

A low-fidelity prototype is a cheap mock-up of a system. It does not use the material that the final product will be made of, and it may not even look like the intended system. However, they still convey the basic components and functionality of the system and can, therefore, still fulfil all the purposes described above. A prototype will use materials such as paper and cardboard rather than electronic screen material with limited functionality, or only represent the functions but not perform any (Preece et al 2019).

The advantages of using low-fidelity prototypes are that they are cheap and simple, and on top of that, they can be produced very quickly. They can therefore easily and without much cost be adapted. They are particularly useful if the designers have just begun and still need to explore different ideas. At this point, you don't want to spend too much on a sophisticated prototype just to find out you have completely missed the point with your design.

Low-fidelity prototypes are not meant to become part of the real system – they are usually thrown away when they have served their purpose (or are kept for sentimental reasons).

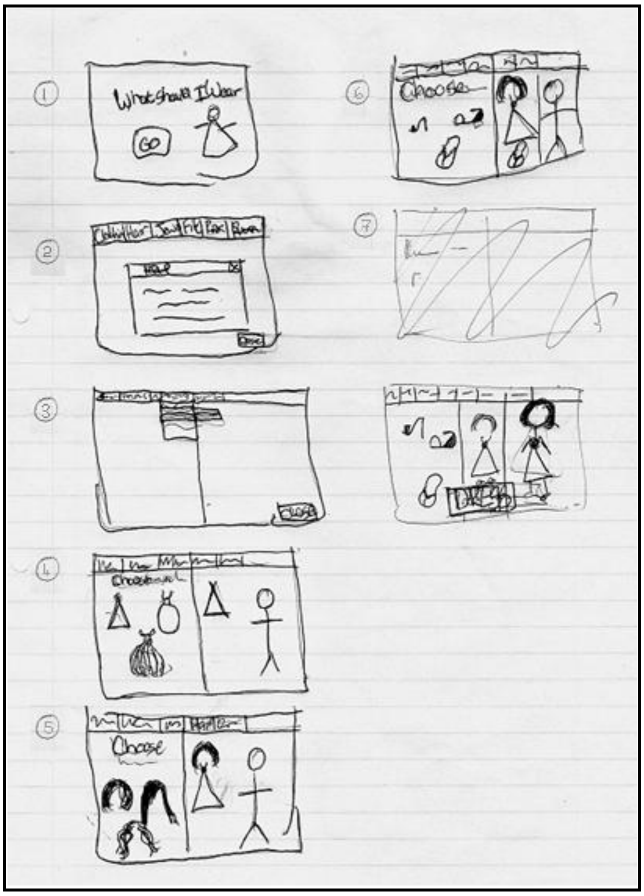


Figure 4.5: A low-fidelity prototype (Preece et al, 2019)

Examples of low-fidelity prototyping are:

- Storyboards – a series of simple images representing the screens show how the user will progress through an application. Figure 4.5 exemplifies such a storyboard created by a twelve-year-old trying to explain her idea for a computer application that helps girls to decide on an outfit for the day. Storyboards can be used in design to demonstrate to consumers how they need to navigate through a task.
- Sketching – more detailed sketches of the interface, including what the icons will look like. This requires better drawing abilities than storyboards.

- Index cards – each card represents a screen or an element of a task, and the user can be taken through different sequences of these.
- Wizard of Oz – here you have a basic software-based prototype, but it still lacks functionality. Although the user uses it as if it had all the functionality, the feedback is provided by someone sitting in a different room at a computer that is connected to the user's computer. This operator makes up responses to the user's actions as they go along.

4.3.1.3 High-fidelity Prototypes

High-fidelity prototypes resemble the final system and usually use the same materials that would be used in the final product. High-fidelity prototypes can be developed by modifying and integrating existing components – hardware and software (Preece et al 2019). This requires software tools and programming skills. Windows programming languages such as Delphi and Visual Basic, are very powerful prototyping tools and can be used by someone with only basic programming skills.

Contrary to low-fidelity prototyping, this method is time-consuming and can be expensive. Because they take long to build, high-fidelity prototypes cannot be adapted easily to changing ideas or requirements. The developers of these prototypes are also more reluctant to change them because of the effort put into them. Because the prototype looks like the final product, the testers tend to comment on superficial aspects such as the look and feel, rather than on the basic functions. A software prototype can create high expectations and make it look as if it is capable of more than it is in reality. One bug in a computer-based prototype system will make it impossible to use.

An advantage of this kind of prototype is that it can gradually develop into the final product. So, the time and resources put into it can be worthwhile. This is called evolutionary prototyping.

Preece et al. (2019) demonstrated and shows a few examples from a high-fidelity version of the low-fidelity prototype, which was developed in Delphi by a twelve-year-old. Although the functionality of this prototype is not real, to the user it seems real.

Preece et al (2019) summarised the advantages and disadvantages of low- versus high-fidelity prototypes as follows:

Table 4.1: Comparison between low- and high-fidelity prototypes (Preece et al 2019)

Type	**Advantages**	**Disadvantages**
Low-fidelity prototype	• Lower development cost • Evaluates multiple design concepts • Useful communication device • Addresses screen layout issues • Useful for identifying marker requirements • Proof of concept	• Limited error checking • Poor detailed specifications to which to code • Facilitator-driven • Limited utility after requirements are established • Limited usefulness for usability tests • Navigational and flow limitations
High-fidelity prototype	• Complete functionality • Fully interactive • User-driven • Clearly defined navigational scheme • Useful for exploration and testing • Look and feel of final product	• More resource-intensive to develop • Time consuming to create • Inefficient for proof-of-concept designs • Not effective for requirements gathering

	• Serves a living specification	

Complete activities 4.3 and 4.4

4.3.2 Conceptual Design

Conceptual design involves turning users' needs and requirements into a conceptual design (Preece et al 2019). Preece et al define a conceptual model as a "high-level description of how a system is organised and operates" (p 51). Shneiderman et al (2014:397) describe conceptual design as "transforming requirements into a conceptual model". It is not a description of the interface, but it gives an idea of what users can do with a system and what concepts they need to be familiar with in order to use it. Using prototypes is one way of getting the conceptual model of the intended system right.

Preece et al (2019) provide the following principles to follow when doing the conceptual design:

- Keep an open mind but always think of the users and their context.
- Discuss the design ideas with all stakeholders as often as possible.
- Use low-fidelity prototyping to get quick feedback.
- Iterate, iterate and iterate – repeat the above again and again until you are sure you have the correct conceptual design.

The conceptual-design process requires the designer to determine how the functions to be performed will be divided between the system and the users; how these functions relate to each other; and what information should be available to perform these functions (Preece et al 2007). With the conceptual model, it is also important to understand how people will interact with the product. The interaction with the product requirements will emerge from the functionality requirements. Preece et al (2019) identified a variety of issues which concept designers should understand. They relate to how to interact with a product:

- Who will the user be?
- What kind of interaction will be used?
- What kind of interface will be used?
- Terminology, metaphors and the application domain need to be understood.

Two important factors in conceptual design are interface metaphors and the interface type.

4.3.2.1 Interface Metaphors

An interface metaphor is an important component of any conceptual model. It provides a structure that is similar to some aspects of a familiar entity, but it also has its own behaviours and properties. We use them to explain something that is hard to grasp by comparing it with something that is familiar and easy to grasp. The Windows desktop is a familiar interface metaphor – the computer screen is like a desktop and the folders and applications are like the things we could in real life have on top of a desk.

Figure 4.6: Using the conquer-the-ogres metaphor to teach multiplication tables
http://gpisdtechhelp.pbworks.com/f/StorybookWeaverDeluxeUserGuide.pdf

The metaphor that is chosen for an interface must fit the task and be suitable for the intended users. Metaphors that appeal to adults may have no meaning for children. The children's game Storybook Weaver uses a book metaphor allowing children to create a cover page and then to fill the pages of the book one by one. Metaphors can also be used to turn something that is potentially boring into an engaging experience. Most children hate learning their multiplication tables. Timez Attack is a children's game that uses the conquer-the-ogres interface metaphor to make this an exciting activity. Figure 4.7 shows a screen from this game: the ogre presents the multiplication sum, and the child (represented by a little green alien) has to build and present the answer within a restricted period of time. Another example is an educational system that teaches 6-year-olds mathematics. A possible metaphor is a classroom with a teacher standing at the blackboard. But given the fact that children form the audience, it is important to consider that they are likely to engage with a metaphor that reminds them of something they like, such as a ball game, the circus, a playground, and so forth (Preece et al 2019).

The purpose of a metaphor is to provide a familiar structure for interaction. The metaphor must be appropriate for the task. The designer should understand the functional requirements, know which bits of the product are likely to cause problems, grasp the metaphors that assist in partially mapping the software and comprehend the real thing upon which the metaphor is based. Only after these are understood, can the metaphor be generated. The user should not be in a position to apply aspects of the metaphor to the system if these aspects are not applicable or will lead to confusion.

Complete activity 4.5

4.3.2.2 Interface Types

When doing the conceptual design, designers should preferably not be influenced by a specific predetermined interface type. Having a specific interface type in mind may stifle the design process and cause potentially good solutions to be overlooked. Another approach would be to reinterpret an initial conceptual design for all the different types of interfaces (in section 4.2, we described eleven) and consider the effect that the change in interface type has on the design. Preece et al (2019) identified four different types of interaction, namely instructing, conversing, manipulating and exploring. These interaction types will depend on the application domain, the kind of product being developed and whether it would be suited to the current design. Any system comes with constraints on the type of interface that can be used.

4.4 Evaluation of Interactive Systems

In the third-level HCI module – INF3720 – evaluation will be dealt with in detail. In this module, we only provide an overview of the process and the methods involved.

Evaluation is a key aspect of human-computer interaction (and of interaction design in particular). It refers to the validation of an interactive system against human-computer interaction requirements (Dix et al 2004). Any design should be assessed, and all systems should be tested to ensure that they meet the user requirements. Evaluation is not a single phase that comes at the end of the design process, but rather an activity that is repeated throughout the design process to provide feedback on the design right from the beginning. For Dix et al (2004) the three main goals of evaluation are to assess the extent of the system's functionality, to assess the effect of the interface on the user and to identify specific problems with the system.

Dix et al (2004) and Williges and Williges (1984) (see figure 4.4) distinguish between formative and summative evaluation.

4.4.1 Formative and Summative Evaluation

Formative evaluation is done early in the design process and continues throughout the design cycle to support design decisions (Dix et al 2004). Low-cost techniques such as low-fidelity prototyping, are suitable for formative evaluation. Early evaluation helps to predict the usability of a product and assesses the designer's understanding of the user requirements.

Often in the design cycle, designers need answers to questions in order to check that their ideas really are what users need or want. In a sense, evaluation meshes closely with design and guides the design by providing feedback. If formative evaluation is to guide development, then it must be conducted at regular intervals during the design cycle.

Formative evaluation is used to prevent problems when users start to operate new systems.

Summative evaluation, on the other hand, is done at the end of the design cycle to test the end product (Dix et al 2004). Its aim is to demonstrate that the completed system fulfils its requirements and to identify problems users have with the system. Usability testing with real users is suitable for summative evaluation. Whereas formative evaluation tends to be exploratory, summative evaluation is often focused on one or two major issues. Interaction designers will be anxious to demonstrate that their systems meet company and international standards as well as the full contractual requirements.

The bottom line for summative evaluation should be to demonstrate that people can actually use the system in their work setting. If sufficient formative evaluation has been performed, then this may be a trivial task. If not, it becomes a critical stage in development.

We can summarise how evaluation fits into the design life cycle as follows:

During the early design stages, evaluation is done to:

- predict the usability of a product or an aspect thereof.
- check if the design team understands the user requirements by evaluating how an existing system is being used in the field
- test ideas quickly and informally as part of envisioning a possible design.

Later in the design process, the focus shifts to:

- identifying user difficulties so that the product can be fine-tuned to meet their needs
- improving an upgrade of the product.

4.4.2 How to Evaluate

Evaluation can be done in laboratories or in the real-life environment where the system will be used. Evaluation methods and the evaluation setting are closely linked. Preece et al (2007) identified five main evaluation approaches which we will discuss in brief.

4.4.2.1 Usability Testing

Usability laboratories with sophisticated audio and video recording facilities and specialised hardware and software for recording and analysing users' behaviour when using a system, are often used for usability testing. Such a laboratory setting gives the evaluator control over the conditions of the study but removes the natural context (with associated interferences) which may be important in the use of the system.

During usability testing, typical users perform selected tasks while their actions are recorded. The evaluator analyses the data collected to judge performance, identifies errors and explains user behaviour. Evaluators should not directly interact with the user in a way that can skew the results. The idea is to derive some measurable observations that can be analysed using statistical techniques. This approach requires specialist skills in HCI.

Usability experiments are usually supplemented with interviews and satisfaction questionnaires.

4.4.2.2 Field Studies

This type of evaluation is done in natural settings. The aim is to understand what users do naturally and to examine how the technology affects them in the real-life environment. The evaluator can be an outsider that observes and records what is happening, or an insider or participant that enters the world of the user to experience the impact of the technology first-hand. Evaluation done in the real environment of use provides the natural context of use, but it may be more difficult to set up the required equipment. Subjects can still be influenced by the presence of researchers in their working environment.

This technique avoids the problem of alienation or irritation that can sometimes be created by the use of interviews and questionnaires. The problem with this approach is that it requires a considerable amount of skill. To enter a working context, observe working practices and have no effect on users' tasks seems to be an impossible aim.

4.4.2.3 Analytical Evaluation

This is either a heuristic evaluation that involves experts who use heuristics and their knowledge of typical users to predict usability problems, or walk-throughs where experts “walk through” typical tasks. The users need not be present, and prototypes can be used in the evaluation. Popular heuristics such as that of Nielsen (2001), were designed for screen-based applications and are inappropriate for technologies such as mobiles and computerised toys.

There are circumstances where a combination of the three techniques will be appropriate. Other evaluation techniques that can be combined with the three methods discussed above, or that can be performed as part of these methods are cooperative and scenario-based evaluation techniques.

4.4.2.4 Cooperative Evaluation Techniques

Cooperative evaluation techniques are particularly useful during the formative stages of design. They are less hypothesis-driven and are an extremely good means of eliciting user feedback on partial implementations.

The approach is extremely simple. The evaluators sit with users while they work their way through a series of tasks. This can occur in the working context or in a quiet room away from the shop floor. Designers can use either low-fidelity prototyping or partial implementations of the final interface. Evaluators are free to talk to users as they perform their tasks, but it is obviously important that they should not be too much of a distraction. If a user requires help, then the designer should offer it and note down the context in which the problem arose. The main point of this exercise is that subjects should vocalise their thoughts as they work with the system. This may seem strange at first, but users quickly adapt. It is important that records are kept of these observations, either by keeping notes or by recording the sessions for later analysis.

This low-cost technique is very effective for providing rough-and-ready feedback. Users feel directly involved in the development process. This often contrasts with the more experimental approaches where users feel constrained by the rules of testing.

A limitation of cooperative evaluation is that it provides qualitative feedback and not measurable results. In other words, the process produces opinions and not numbers. Cooperative evaluation is extremely ineffective if designers are unaware of the political and other pressures that might bias a user’s responses (that is, might influence them either positively or negatively).

4.4.2.5 Scenario-Based Evaluation

Scenarios are informal narrative descriptions of possible situations. In interface design, it is a sample trace of interaction. This approach forces designers to identify key tasks in the requirements gathering stage of design. As design progresses, these tasks are used to form a case book (containing standard tests) against which any potential interface is assessed. Evaluation continues by showing the user what it would be like to complete these standard tests using each of the interfaces. Typically, users are asked to comment on the proposed design in an informal way. This can be done by presenting them with sketches or simple mock-ups of the final system.

The benefit of scenarios is that different design options can be evaluated against a common test suite. Users are in a good position to provide focused feedback about the use of the system to perform critical tasks. Direct comparisons can be made between the alternative designs. Scenarios also have the advantage that they help to identify and test design ideas early on.

The problem with this approach is that it can focus designers' attention upon a small selection of tasks. Some application functionality may remain untested, and users become all too familiar with a small set of examples. A further limitation is that it is difficult to derive measurable data from the use of scenario-based techniques. In order to derive such measurable data, scenario-based techniques must be used in conjunction with other approaches such as usability testing.

Complete activity 4.5

4.4.2.6 Query techniques

The two main types of query techniques are interviews and questionnaires.

Interviews

Using interviews for evaluation purposes involves asking users questions about their experiences with the system under evaluation. For an interview to be effective, the interviewer has to plan it carefully by preparing specific questions or making a list of topics to address. This will help to focus the interview and to keep the issues discussed with different users consistent. However, the interview should not be overly structured so that the interviewer can easily adapt the questioning to suit the specific user. Structured interviews are easier to conduct and analyse than flexible interviews. On the other hand, they may miss important details relating to the user's experience which a more flexible interview would have picked up. A good compromise is a semi-structured interview which is based on leading questions but has the flexibility to investigate promising or unanticipated directions.

In general, the advantages of interviews are that the evaluator can vary the level of questioning to suit the context, and that the evaluator can probe the user for more information on relevant issues which arise spontaneously during the interview.

Questionnaires

A questionnaire consists of fixed questions relating to interaction with the system being evaluated. When used for the purpose of evaluation, users complete the questionnaire after completing a given task or set of tasks. This technique is less flexible than interviews, but it has several advantages. A larger number of users can be included in the evaluation as it is less time-consuming and labour intensive, and the results can be analysed more rigorously. Different styles of questions can be used, namely open-ended questions (which allows the user to give his or her opinion freely) as well as closed questions such as multiple-choice or ranked questions. Questionnaires can be completed in fixed sessions, or they can also be administered independent of time and place, and without the presence of an evaluator.

4.4.2.7 Heuristic evaluation

This evaluation technique was developed by Jakob Nielsen and his colleagues (Nielsen 1994). Applying heuristic evaluation means that user interface design experts evaluate the user interface according to usability principles known as heuristics. The process of heuristic evaluation involves three steps:

1. briefing: experts are told what to do.
2. evaluation: each expert spends a few hours taking at least two passes through the interface, using the heuristics to identify problems.
3. debriefing: experts meet to discuss their evaluations, prioritise problems and suggest solutions.

The advantage of heuristic evaluation is that there are fewer practical and ethical issues to take into account as users are not involved. A disadvantage is that the experts often identify problems that are not really problems. This suggests that heuristic evaluation should preferably be used along with other techniques and that several evaluators should take part in the evaluation.

We end this section with a list of Nielsen's evaluation heuristics formulated as questions:

1. How good is the visibility of system status?

2. Is there a clear match between the system and the real world?

3. Do users have control when needed and are they free to explore when necessary?

4. Does the user interface display consistency and adherence to standards?

5. Does the interface help users to recognise and diagnose errors and to recover from them?

6. How good is the error prevention?

7. Does the interface rely on recognition rather than on recall?

8. How flexible and efficient is it to use?

9. How good is the interface in terms of aesthetics and minimalist (clear and simple) design?

10. Is adequate help and documentation available?

Complete activity 4.6

4.5 Conclusion

This lesson provided a mixed bag of information on interaction design. We described a range of interface types that are currently available. We also looked at techniques that designers should use during the interaction design process, specifically prototyping and conceptual design.

Evaluation plays a major role in interaction design. Without some form of evaluation, it is impossible to know if the system satisfies the needs of the users and to determine how well it fits the physical, social and organisational context in which it will be used. This lesson tool introduced evaluation methods that can be used in the design of interactive systems.

4.6 Activities

ACTIVITY 4.1
DStv Customer Services uses a speech interface on their telephone enquiry system. For this activity, you have to phone them. Your aim with this phone call is to find out what packages they offer and what the cost of each package is. Their number is:

083 900 DSTV (3788).

Now describe your experience with the speech interface by pointing out specific problems with and good aspects of the interface.

ACTIVITY 4.2

Complete the following table. Do not rely only on the information given above. Try and find examples, advantages and problems not mentioned here.

Interface Type	**Description**	**Advantages**	**Problems**	**Application examples**
Web-based				
Speech				
Pen, gesture, touchscreen				
Mobile				
Multimodal				
Shareable				
Tangible				
Augmented and mixed reality				
Wearable				
Robotic				

ACTIVITY 4.3

Compare low- and high-fidelity prototyping by completing the following table.

Type of prototype	**Advantages**	**Disadvantages**
Low fidelity		

High fidelity		

ACTIVITY 4.4

Suppose you have been asked to design a web-based system for renting online DVDs. The system allows users to

- browse the available movies
- select one or more that they want to watch
- make a credit card payment
- receive the access keys for the selected movies via SMS

The idea is that the user can watch movies online (within a 24-hour period) from the store's website by providing the access keys.

Create a low-fidelity prototype (in the form of a storyboard) for this system.

ACTIVITY 4.5

1. Identify any interface metaphor in a system with which you are familiar. Answer the following questions about this metaphor:
    a. How does it give structure to the interaction process?
    b. How much of the metaphor is relevant to the use of the system? (In other words, are all aspects of the metaphor relevant or only some? Can those that are not relevant lead to confusion in the interface?)
    c. Is the metaphor easy to interpret? (Will users easily understand the metaphor?)
    d. How extensible is the metaphor? (If the application is extended, will unused aspects of the metaphor be applicable?)

2. Identify and describe a suitable interface metaphor for the online DVD renting application of activity 4.4.

ACTIVITY 4.6

Use Nielsen's heuristics to evaluate the assignment submission pages of the myUnisa website. Your evaluation report should provide answers to each of the 10 (ten) questions and should include examples of interface elements where applicable.

2.5 Lesson 5: Social Aspects of Computer Use

The content of this lesson is as follows:

CONTENTS

LESSON TOOL 5 – OUTCOMES

After studying this lesson, you should:

- understand the impact of computer technologies on society and on
  - e-commerce and business
  - our working lives
  - education
  - information processing
  - privacy and security
- be able to identify the social problems associated with advanced computer technology
- be able to demonstrate a comprehensive understanding of the impact of social network websites – their advantages and disadvantages
- be able to define the term "digital divide" and describe its causes and how it can be addressed

5.1 Introduction

Our world and all aspects of life have become inundated with computer technologies. On the one hand, we are empowered by these technologies, and they have improved our quality of life. On the other hand, they invade our privacy and widen the gap between the rich and the poor. This last lesson tool of the study guide introduces you to the wider social implications of computer technology.

5.2 The Impact of Information Technology on Society

In the discussion of the ways in which computers and technology in general influence how we live and relate to our environment, we will focus on the following aspects:

- e-commerce and business
- our working lives
- education
- information processing
- problems associated with advanced computer technology

5.2.1 Business and E-commerce

Computer technology and the internet, in particular, have completely changed the way commercial companies do business and banks function. Physical distance no longer puts a restriction on the way businesses are structured. A company can have divisions located in different countries and responsibilities can be divided between countries on the basis of the skill and expertise located there. By dividing labour across countries, production companies can now have people working 24 hours of the day. Manufacturing can be allocated to where it can be done most efficiently.

Network infrastructure and the online availability of services and information have made salesclerks, stockbrokers and travel agents redundant. In the past, they were the main information links between companies and clients. Nowadays web-based services have taken over this function.

The fact that products such as software and music can be "shipped" electronically has reduced the need for distribution and shipping companies. Digital music has completely changed the music industry worldwide. From a music lover perspective, this is a positive development because music can be bought over the web. No album is difficult to come by anymore. Even an obscure band such as "Indie", can sell its music online. Another advantage is that you can buy individual songs – no need to buy the whole album if you are after only one or two tracks. From record companies' and producers' point of view, this change has forced them to rethink their business models. There is now much more competition – artists can easily market and sell their music on the internet without the help of a record company. Advances in computer technology have also impacted the recording and production of music. High-quality recordings can now be made using mobile studios and complete sound mixing and editing systems now come in the form of software applications (see figure 5.1). The cost of making an album, if you have the technological skills, is much reduced.

Figure 5.1: An Apple Garageband screen-based mixing interface
(https://apps.apple.com/gb/app/garageband/id682658836?ls=1&mt=12) [2]

Besides the exclusion of shipping and distribution, there are numerous ways in which replacing a physical business with an online one reduces cost:

- No physical store needs to be set up and maintained.
- Order placement and execution is simplified.
- 24-hour customer support is provided.
- Staffing requirements are reduced.
- A retail business does not need to carry the inventory of a physical store.
- There are no restrictions on retail hours.

The main cost now lies with the setting up and maintenance of a store website. Since the website is always “open” and can be reached by millions of potential customers, spending on the usability and appearance of the site is justified.

The shipping of large and expensive products bought over great physical distances can increase their cost; but still, these products may not be available to the customer in any other way.

---

[2] Although every effort has been made to trace the copyright holders, this has not always been possible. Should any infringement have occurred, the publisher apologises, and undertakes to amend the omission in the event of a reprint.

Unfortunately, e-commerce creates opportunities for fraud and theft. Measures to prevent this and insurance against it may mean added costs for business. It is quite easy to get unlawful access to digital music on the internet. The same applies to the movie business. According to *New Scientist* (13 March 2010), there were 7,9 million pirated downloads of the movie The *Hurt Locke*r before it won six Oscar awards in 2010. In 2009, almost 33% of all internet users in the United Kingdom were using unofficial online sources of music (*New Scientist*, 5 December 2009).

5.2.2 Working Life

5.2.2.1 Communication and Groupware

The most pronounced changes that technology has brought into the workplace are the electronic mechanisms for communication such as e-mail and Skype. They allow workers to correspond cheaply and instantly over great distances. Collaborative work can be done by people who reside in different countries and who might never meet face to face.

Web 2.0 technology is commonly used by organisations to support collaborative work (this utilisation of Web 2.0 within a secure environment developed into what is sometimes called Enterprise 2.0). The historical term used for collaboration through computer technology is Computer-Supported Cooperative Support (CSCW). CSCW is concerned with the principles according to which computer technology support communication and group work. The physical systems through which CSCW manifests are collectively referred to as Groupware. These systems never gained widespread popularity owing to technical and design issues such as hardware and operating system incompatibilities, and the inability to understand the effects of how groups and organisations function. Some of the specific problems are:

- Synchronous and asynchronous systems: It may be difficult for users to know exactly who else is using the system. Synchronous means "at the same time". Asynchronous means "at different times" or independent. You get both types of CSCW systems. For example, a system that supports collaboration between groups in Australia and South Africa will be asynchronous. Time differences mean that there would only be small periods of time when they would both be working on the system. If the application is asynchronous, then many of the problems of contention (see below) do not arise.

- Contention: This occurs when two or more users want to gain access to a resource that cannot be shared. For example, it may not be possible for people to work on exactly the same piece of text at the same time.

- Interference: This arises when one user frustrates another by getting in their way. For example, one person might want to move a piece of text while another attempts to edit it. Similarly, one user might want the others in a group to vote on a decision while another user might want to continue the discussion.

5.2.2.2 Access

Easy access to information has also impacted on the work environment. Employees are empowered by the electronic availability of company reports and policies on internal networks. Further empowerment comes with e-mail that makes it easier for employees at lower levels to communicate with their superiors. The problems of intimidating face-to-face encounters are reduced, and employees can communicate with their superiors without the fear of intrusion (as with personal meetings and telephone calls). In other words, managers have become more accessible.

5.2.2.3 Office Hours and Location

Technology also affects the way companies think about office hours and office space. Mobile technology allows people to do their work anywhere, any time, and a centralised office may not be important any longer. This will benefit companies who can cut down on office space and it will be advantageous to employees who will have more flexible work hours. People do not need to live close to the office.

On the downside, it has become difficult to separate one's personal and working lives. Being connected at all times heightens the need for skills such as prioritising, focusing and working without interruption.

5.2.3 Education

As UNISA students you have first-hand experience of how advances in technology have influenced education. A learning management system such as myUnisa, makes it possible for students to communicate with their lecturers, participate in discussions with their fellow students, submit their assignments (at the last minute), check their examination and assignment schedules, and much more. Lecturers can send urgent notices to students via SMS.

You can download an electronic copy of this study guide from the INF1520 tutorial matter page, store it on your mobile phone, iBook or Notebook, and read it while lying next to the pool.

There is a vast number of educational resources available on the internet. Renowned universities such as MIT in Boston, make courseware available on the internet for free (see the MITOpenCourseware site at http://ocw.mit.edu/index.htm).

Complete activity 5.1

Unfortunately, there is probably more unreliable information of questionable quality available than trustworthy academic sources. Children should therefore be taught from an early age to distinguish between reliable and questionable information.

A current hot topic in HCI research is m-learning (mobile learning). It involves the use of cellphones and other mobile devices as delivery mechanisms in education. Most students today own or have access to a cellphone, although some still do not have easy access to computers. Cellphones are therefore an ideal platform to distribute learning material to students. But it introduces a new design challenge: How does one present learning material on such a small display?

There are numerous problems associated with these trends in education. The younger generation (i.e., students) may expect more from technology than what the older generation (their lecturers) feel comfortable with or may think possible. Another problem has to do with the digital divide discussed in section 5.3 below – not everybody has access to the technological resources required for e-learning and m-learning. Some of the activities in this lesson tool require access to fast internet connections and sufficient bandwidth. Not all students will have this.

Complete activity 5.2

5.2.4 Information Processing

Supercomputers have made it possible to process huge amounts of data in a relatively short time. It is now possible to develop computer models of complex systems. Climate modelling is one field that relies on the computing power available today. This involves the numerical representation of aspects of the climate system that makes it possible to predict future weather patterns. Just a few years ago, it was quite difficult to simulate weather conditions for any extended time. Nowadays, merely by clustering high-speed processors to perform the calculations simultaneously, a simulation of a full year's weather takes under three weeks to complete (Lepage, Qiu & Sifton 2006).

The human genome project provides another example of research that required this kind of computing power. Without this capacity, the aim of establishing a human DNA sequence could not have been reached.

The social advantages of being able to model complex systems include a better (Muglia 2010):

- understanding of pandemics, contagion and global health trends
- prediction of the impact of climate change on the environment, the economy and on humans in general
- prediction of natural disasters and their impact so that effective response plans can be set up

Google Earth (earth.google.com) enables us to "visit" foreign places. The system's viewing interface allows users to get a bird's eye view of just about any location in the world. The system uses a search engine, satellite images, maps and 3D buildings. One can zoom in to get close to a specific building or other location and also rotate the view to change the 3D perspective. Using Google Earth requires a high bandwidth and a computer with a relatively high processing power.

Complete activity 5.3

5.2.5 Problems Associated with the "Information Age"

The "myth of the cyborg" has had an important impact upon popular attitudes towards the safety of technology. This myth has been popularised by Hollywood films such as the Terminator series. It is, however, part of a longer stream of literature. These earlier sources include writers such as HG Wells and Mary Shelley. They were less concerned about the nature of the technology itself than about the (potentially subservient) relationship between people and the machines that they create. In modern times, complex computer-controlled systems have increased the relevance of this relationship. However, the threat may be less spectacular than that portrayed in the Terminator series. We are not threatened today by a malign supercomputer seeking to destroy humanity. However, our everyday safety is almost entirely dependent on information technology. There is a very real sense of the fact that we are all cyborgs. We are all dependent on machines to survive.

In this section, we briefly look at three specific problems that can be linked to the almost immeasurable availability of information, namely the problem of keeping information private and secure, the difficulty of filtering reliable information and our dependence on technology.

5.2.5.1 Privacy and Security Issues

Complete activity 5.4

The increase in computing and communications power poses a threat to both the public and the private sector. Governments, banks and private institutions keep electronic data on individuals, and it is not always clear how safe that information is. Many people aren't even aware of the fact that information about them is kept in databases that may not be completely secure. If you are a DStv subscriber, your decoder or PVR is connected to a network through which information such as your daily viewing patterns and what programmes you record, can be accessed. The records of calls and SMS messages of cellphone users are accessible through the mobile network they are subscribed to. The personal information that people make available on social network sites such as Facebook, is an indication that they are not concerned about the fact that this information can be accessed by people with dishonest intentions.

People do not see digital data and physical information in the same light. Because digital data is not tangible, they think it is less important to secure it. This is a dangerous assumption. The biggest difference between digital and physical sources of information is that it is far easier to copy and forge digital data. It is also easy to find information that users do not protect.

An annoying breach of e-mail users' privacy comes in the form of spam – unsolicited mass mail that is sent to millions of users daily. *New Scientist* magazine (27 February 2010) reported on a study done over a period of one month in 2008 regarding pharmaceutical spam. The following was found: Of about 35 million spam messages that were sent, only 8.2 million reached a mail server. Only 10 500 recipients clicked on the link in the mail and only 28 of them actually bought products. The 35 million messages made up only 1.5% of what one spam botnet (a network of remotely controlled computers) produced in a month. Extrapolating from this information is that specific spam botnet generated around $3,5 million in pharmaceutical sales in 2008. So, although an insignificant number of recipients respond to spam messages, it is still worth the trouble for spammers.

The growing value of the information being stored and transferred across the world's computer networks also increases the importance of security. In the past, organisations could preserve their security by denying all external access to their systems. The increasing use of the web to advertise and sell products means that more commercial systems are hooking up to the internet. The growing communication opportunities provided by electronic mail and social networks have also encouraged greater interconnection. All these factors increase the stakes for malicious and criminal users. Electronic fund transfers and commercially sensitive e-mail messages are tempting targets.

The technological sophistication of the general population is on the increase. This means that more and more people have the knowledge and ability to beat the system, and that commercial and government organisations must continually try to stay one step ahead of the people who "hack" or "crack" into their systems.

Software that is developed for the purpose of doing harm or gaining unlawful access to information is referred to as "malware".

This generic term includes intrusive code such as:

- Trojan horses: This form of attack refers to the ruse of war used by the Greeks to invade the city of Troy by hiding inside a wooden “gift horse”. In computer terms, a malicious piece of code is hidden inside a program that appears to offer other facilities. For example, a file named “Really_exciting_game” will contain a rather boring game but also a program that attempts to access your password file. Once the program has obtained a list of usernames and passwords, it may write them to a file that is visible to the attacker. From then on the gates are open, and the system is insecure. An alternative approach would be for the program to continue to run after a user thinks he/she has quit. The intruder might then be able to use the still running program to gain access to the user’s files and resources.

- Time bombs: These are planted as a means of retaliation by employees who are disgruntled because they have been dismissed. For example, a program might be scheduled to run once every month. The code would check payroll records to see if the disgruntled employee’s name is on it. If it is, nothing happens; if it is not, the program takes some malicious action. For example, it might delete the rest of the payroll or move money to another account. Such programs constitute a major security breach because they require access to personnel data and the ability to take some malicious action. The long-term consequences may be less severe than that of a Trojan horse because the system may still be secure in spite of the damage caused.
- Worms: These self-replicating programs represent a major threat because they will gradually consume more and more of your resources. Your system will slowly grind to a halt and all useful work will be squeezed out. This useful work will include attempts to halt the growth of the worm. It takes considerable technological sophistication to write a worm. You must first gain a foothold in the target machine. Then you have to create a copy that can be compiled and executed on that host. This copy must then generate another copy, and another one, and so on. The main difference between a virus and a worm is that a worm does not need a host to cause harm.

The producers of malware are referred to as “black hats”, “hackers” or “crackers”. (Note: some legitimate programmers call themselves “hackers” because they hack the code into shape. The media often refer to people who attack computer systems as “hackers”. Many programmers find this a misnomer and prefer the term “crackers” for people who attack systems.)

It is generally assumed that most security violations in large organisations come from within and are the result of either malicious actions or carelessness. The former takes the form of industrial or military espionage. The latter occurs when someone inadvertently leaves a flash drive or printouts in a public place.

As the greatest security threats come from within an organisation, it follows that many companies have clear rules of disclosure. These specify what can and what cannot be revealed to outside organisations. They extend to the sort of access that may be granted to the company’s computer systems. A particular concern here is the repair facilities that may be provided for machines that contain sensitive data. One of the most effective means to breach security is to act as a repair technician and copy the disks of any machine that you are working on. In fact, security is based on a “transitive closure” of the people you trust. This basically means that if you pass information on to someone you trust, then you’d better be sure that you trust all the people whom this person trusts and so on. If they pass your information on to someone else, then you have to trust all the people whom this person trusts.

5.2.5.2 Information Overload

New information leads to new inventions and eventually contributes to the evolution of humankind. Our existence depends on knowledge; so, we are naturally predisposed to crave information. The internet has taken this to a new level as we now have more information at our disposal than is good for us. People spend a great amount of time searching through and taking in irrelevant or useless information just because it is there (Konsbruck [sa]). Much of what appears on the internet is incomplete, unsubstantiated and incorrect. Even worse, people now have access to harmful information such as:

- instructions to build explosive devices
- the genome of the influenza virus that killed over 50 million people in 1918 (Parsons 2010) (With the right tools anyone can reconstruct it.)
- political propaganda
- pornographic and violent video material

There is a need for research to determine how people judge the credibility of information and for systems that help people to survive the information overload (Konsbruck [sa]). Being able to filter information is a skill that has become extremely important, and mechanisms have to be developed to help those who struggle with this.

5.2.5.3 Dependence on Technology

Modern society is almost entirely supported by information technology. Although there are individuals and even communities that still function without access to technology infrastructure, society has, for the most part, reached a point of complete dependence on technology. The risk of this dependence is that the breakdown of technological infrastructure will lead to a serious disruption of economic and social systems (Konsbruck [sa]). We cannot live without mobile phone technology, credit data systems, electronic money transfer systems, and the like. If the world's computer systems fail, so would our food and power distribution networks.

An incorrect notice that went out to Unisa staff about the unavailability of web-based services, including myUnisa, which seems to be a relatively small problem, that can have far-reaching consequences. Students who rely on the electronic submission of assignments may miss the due dates of assignments. If such a problem persists, the university may be compelled to give a general extension on assignment due dates, interfering with the tuition schedules of individual modules. Assignments may not be marked and returned to students in time before the examination starts.

5.3 Social Networking Technologies

In lesson tool 1 (section 1.2.5.4), we briefly introduced social network websites (also referred to as online communities). In this section, we take that discussion further by looking at other types of social networking tools such as blogs and instant-messaging systems.

5.3.1 Chat Rooms

Chat rooms are locations on the internet where people meet to have online conversations in real time. They are usually open for anyone to join, and they often relate to a specific topic (for example, a support group for people with a specific medical condition). The messages are normally visible to all current visitors, but users can participate in private conversations not observable by others. Some chat rooms allow users to publish their photographs and personal information or use web cameras. There may be rules for participation and the content may be monitored, but many have no restrictions on what may be discussed or in what kind of language. There is a danger that young users can be targeted by sexual predators. Ybarra and Mitchell (2007) report that chat rooms have been losing popularity among teenagers since 2000 because many of them regard these rooms as unpleasant places.

5.3.2 Instant messaging (IM)

The South African MXit system is an example of an IM system. It is a real-time communication tool that allows two or more users who are connected to the system to interact with each other synchronously. IMs are different from chat rooms in the sense that the sender must know the username of the recipient to send a message. Some IM services have a searchable member directory where users can add information so that other users of the system can identify them to send a message (Ybarra and Mitchell, 2007). Privacy settings make it possible to block out messages from unknown users or messages from specific individuals.

5.3.3 Blogs

Blogs are like online journals. Individuals use them as diaries or to comment on specific topics. Some allow readers to post responses. Blogs are popular amongst children. In 2009, 24% of all children between the ages of 9 and 16 in the United Kingdom had their own blogs (*New Scientist* 12 December 2009).

5.3.4 Social Networking Sites

Social networking sites are sometimes referred to as web communities or online communities. These sites integrate all of the tools above. Users create profiles with descriptive personal information and photographs and write messages on message boards or walls (as in a blog). Profiles are interconnected through explicitly declared friend relationships (Caverlee & Webb 2008). Users communicate through different kinds of messaging mechanisms. Communication can be synchronous (like “chatting”) or asynchronous (like blogs or e-mails). Users can limit access to their profiles through privacy settings or they can leave it public and allow anybody to view their information.

It has been possible to publish personal information on the WWW for a long time. The uncomplicated mechanisms that social networking sites provide to do this have now caused an explosion of detailed personal data on the internet. The success of social networking sites can be attributed to the fact that they provide a standardised, centralised, easy and free way to create an internet presence (Jones & Soltren 2005).

A social network in the general sense refers to all the people someone has a social relationship with. Online social networks, however, include “friends” whom the user has never met and has no other link to besides the fact that they appear as friends on their profile. Users of social network sites typically communicate directly with a few individuals in their friend lists (Huberman, Romero & Wu 2008).

Privacy is an important issue linked to online social networks. It seems that, for some time after Facebook emerged, users were ignorant of the consequences of excessive disclosure of personal information (Jones & Soltren 2005). Caverlee and Webb (2008) report that there is a steady growth in the use of privacy settings by new members on MySpace, one of the most popular online communities. This indicates that people are becoming more aware of the privacy risks associated with these sites.

Online social networking has influenced the way people interact and relate to one another. It is a cheap and easy way for family and friends who are physically removed to stay in touch. Introverts who in the physical world would seldom meet new people and make friends, can now build online relationships in the “safe” environment of a web-based community. It can, unfortunately, become an addiction that compels people to constantly check for Facebook updates or Twitter messages. I recently spent a weekend in Zambia on the banks of the breathtaking Kariba Lake with some friends. In our party were three women in their late twenties and early thirties who were inseparable from Facebook. For me, the weekend inadvertently became an HCI research project, an observation of the three women's interaction with technology. They were completely incapable of appreciating the beauty of the lake and its surroundings by just looking at it. Every view had to be photographed and immediately loaded onto a Facebook photo album. They then gathered around a laptop and enjoyed their holiday by browsing the photographs of it. It was obvious that they would have been at a complete loss if their cellphones and notebook computer were taken away.

Some advantages of social networking sites include:

1. the low cost of creating a web presence
2. making personal connections by, for example, searching for people who share your interests or becoming friends with friends of friends. You can also reconnect with long-lost friends. For many people social network sites are their primary mechanism to find a date
3. connecting families by, for example, allowing family members to connect with one another when they stay far apart or in different countries
4. making connections for career purposes – it is quite easy to identify people who work in your field by searching through their profiles
5. businesses getting additional information on someone before employing them (It is a way to find out if people have lied in their applications or CVs.)

Some disadvantages of social networking sites include:

1. a lack of anonymity or privacy
2. identity theft – some people place enough information on these sites to allow others to get all the necessary information to assume their identity
3. a loss of working time – some companies block access to these sites during office hours
4. mining of users' data for advertising purposes
5. cyberbullying – it is much easier to harass someone through an online network than it is in the real world
6. cyberstalking
7. inappropriate content such as political propaganda (Countries such as Syria, China, Iran, and Vietnam have banned the use of Facebook.)

Complete activity 5.5

5.4 The Digital Divide

The digital divide refers to unequal access to technology that subdivide people into those who have it and those who do not (Attewell, Suazo-Garcia & Battle 2003). Some of the contributing factors to the digital divide are financial constraints, the lack of skills, unavailability of basic infrastructure (e.g., electricity) and carelessly designed systems. In developing countries where technology and internet access are relatively widely available, there are still problems with fast internet access. Some problems are associated with online interaction, especially where bandwidth is limited. Internet content providers often do not show consideration for these limitations and will not compromise, for example, on the use of sound and graphics that can take a long time to download. This means that a large part of the potential user population is unable to use internet-based applications. Literacy levels also contribute to the digital divide.

A lack of cognitive resources is an important contributor to the digital divide (Wilson 2006). Interacting with computers requires basic skills to recognise the need for information, to find the information, to process and evaluate the information for its appropriateness, and to apply it in a meaningful way.

The digital divide is not only a reflection of the separation between developed and developing economies. It can also exist among population groups within the same nation. In the United States, white and Asian people are at least 20% more likely to own computers than black and Hispanic people (Cooper & Kugler 2009). In 2001 only 2% of black households in South Africa had computers compared to 46% of white households (Statistics South Africa 2001).

Complete activity 5.6

Many attempts are being made to bridge the divide, for example, the Digital Doorway project (Meraka Institute, accessed 23 Oct 2007) in South Africa, MIT's One Laptop Per Child project (MIT, Accessed 23 Oct 2007) and the Hole-in-the-Wall project in India (Mitra, 2003). The Digital Doorway project is a joint initiative by the Department of Science and Technology and the Meraka Institute of the CSIR. It focuses on providing computers in underprivileged communities in South Africa. Digital Doorways are non-standard computer systems housed in rugged, custom-designed kiosks with multiple terminals that can be accessed simultaneously by users (see figure 5.3). The robust housing and metal keyboard protect the system against vandalism. The aim is to promote computer literacy through unassisted learning (Cambridge 2008, Gush, De Villiers, Smith & Cambridge, In Press) by installing the computers at schools, police stations and community centres in underprivileged communities across South Africa. By mid-2010 a total of 206 Digital Doorways had been deployed across the country.

Complete activity 5.7

5.5 Conclusion

Many books have been written on social issues in human-computer interaction and the impact of technology on society. So, it is difficult to reduce these topics to a discussion in one lesson tool of a study guide. Our aim was to give you an idea of the profound effects of technological advancement on society. We only touched on aspects such as privacy and security and did not even mention other important issues such as ethics and intellectual property. We do hope that we have stirred your interest and that you will, as future designers or IT managers, keep yourself informed of the pervasive impact that computers have on our world.

5.6 Activities

ACTIVITY 5.1

David Kieras is a world famous researcher in HCI who specialises in the psychological aspects of the field. He is a professor in the Electrical Engineering and Computer Science Department at the University of Michigan. The internet makes it possible to attend lectures of academics of his stature from anywhere in the world.

If you have access to a fast internet connection, go to the website http://videolectures.net/chi08_kieras_phc/. Watch the video of a lecture by David

Kieras titled “Psychology in Human-Computer Interaction”.

List five (5) important aspects of HCI that you have learnt from this lecture. (You will need this list in the next activity.)

ACTIVITY 5.2

Go to the INF1520 discussion forum on myUnisa and do the following:

1. Post a message with at least one suggestion on how to improve the teaching of this module using technology. Use “Activity 5.2” as the subject.

2. Post a message with the list of five things you learnt from the Kieras lecture (see activity 5.1). Use “The Kieras lecture” as the subject. Some students may not have been able to do activity 5.1 due to bandwidth problems. If those students who were able to watch the lecture post their insights on the INF1520 discussion forum, then the students who could not watch it will also get the benefit of the activity.

ACTIVITY 5.3

Only students who have fast internet access will be able to do this activity. You need to access Google Earth (http://earth.google.com) and download and install the software if you do not already have access to it.

Suppose you are offered a scholarship to work in the USA at the University of Maryland, College Park for a year. Your spouse and two children will accompany you. You want to live relatively close to the university and a primary and a secondary school for your children.

Use Google Earth to find a good location where you can rent a home. Identify a specific address.

ACTIVITY 5.4

Think carefully about the places on the internet and other networks (for example, mobile networks, electronic banking data and Facebook) where information about you is possibly stored. Make a list of everything a clever hacker can find out about you by accessing these data sources.

ACTIVITY 5.5

List five (5) more advantages and five (5) disadvantages of social networking sites. Use examples from your own experience using these sites.

ACTIVITY 5.6

UniNet is a South African telecommunications company. They have made a documentary on the project entitled UniFi Knysna through which the whole of Knysna was supplied with Wi-Fi (Wireless Internet). Watch the video at the following link: http://www.youtube.com/watch?v=TpxCm8Snt9I.

ACTIVITY 5.7

Do research on the internet about the One Laptop Per Child (OLPC) project. Write a one-page essay on this project that includes at least the following information:

- Who initiated it and when?
- What does the project involve?
- Where has it been deployed?
- How successful is it?
- What problems are associated with the project?

REFERENCES

ACM SIGCHI (n.d.) *Curricula for Human-Computer Interaction*, http://www.sigchi.org/, accessed 24 June 2010.

ANTLE, A. (2007) Designing Tangibles for Children: What Designers Need to Know. *CHI 2007.* San Jose, California, USA, ACM Press.

ATTEWELL, P., SUAZO-GARCIA, B. & BATTLE, J. (2003) Computers and Young Children: Social Benefit or Social Problem? *Social Forces*, 82**,** 277–296.

BAECKER, R. M. & BUXTON, W. A. S. (Eds.) (1987) *Readings in Human-Computer Interaction: A Multi-Disciplinary Approach*, Los Altos CA, Morgan Kaufman.

BARTLETT, B., ESTIVILL-CASTRO, V. & SEYMON, S. (2004) Dogs or Robots: Why do Children see them as Robotic Pets rather than Canine Machines? In COCKBURN, A. (Ed.) *Conference on Research and Practice in Information Technology.*

CAMBRIDGE, G. (2008) Digital Doorway: Enriching your Mind. *2nd CSIR Biennial Conference.*

CARROLL, J. M. (2003) Introduction: Toward a Multidisciplinary Science of Human-Computer Interaction. In CARROL, J. M. (Ed.) *HCI Models, Theories and Frameworks.* San Francisco, Morgan Kaufman Publishers.

CARROLL, J. M. (2009) *Encyclopedia entry on Human Computer Interaction (HCI)*, http://www.interaction-design.org/encyclopedia/human_computer_interaction_hci.html, accessed 22 June 2010.

CAVERLEE, J. & WEBB, S. (2008) *A Large-Scale Study of MySpace: Observations and Implications for Online Social Networks*, accessed 1 July 2010.

COOPER, J. & KUGLER, M. B. (2009) The Digital Divide: The Role of Gender in Human-Computer Interaction. In SEARS, A. & JACKO, J. A. (Eds.) *Human-Computer Interaction: Designing for Diverse Users and Domains.* Florida, Taylor & Francis Group, LLC.

DARZENTAS, J. & MIESENBERGER, K. (2005) Design for All in Information Technology: A universal Concern. In ANDERSON, K. V., DEBENHAM, J. & WAGNER, R. (Eds.) *Lecture notes in Computer Science.* Springer-Verlag.

DEIGN, J., ROGERS, Y. & SHARP, H. (2007) *Interaction Design: Beyond Human-Computer Interaction*, John Wiley & Sons, Inc.

DIX, A., FINLAY, J., ABOWD, G. & BEAL, R. (2004) *Human-Computer Interaction*, Pearson Education Ltd.

DONG, Y. & LEE, K. (2008) A Cross-Cultural Comparative Study of Users' Perceptions of a Webpage: With a Focus on the Cognitive Styles of Chinese, Koreans and Americans. *International Journal of Design,* 2.

DRUIN, A. (1996) CHIKids: A Common Ground for Kids and Adults. In TAUBER, M. J. (Ed.) *Common Ground: CHI 96 Conference Proceedings.* ACM.

GAME VORTEX (2008) *EyeToy Play 2,* www.psillustrated.com/gamevortex/soft_rev.php/2686, accessed 22 January 2008.

GATHERCOLE, S. E. (2002) Memory Development During the Childhood Years. In BADDALEY, A. D., KOPELMAN, M. D. & WILSON, B. A. (Eds.) *The Handbook of Memory Disorders.* Second ed., John Wiley & Sons, Ltd.

GUSH, K., DE VILLIERS, M. R., SMITH, R. & CAMBRIDGE, G. (In Press) Digital Doorways. In STEYN, J., BELLE, J. & VILLENEUVA, E. M. (Eds.) *Development Informatics and Regional Information Technologies: Theory, Practice and the Digital Divide.* Pennsylvania, IGI.

HARPER, R., RODDEN, T., ROGERS, Y. & SELLEN, A. (Eds.) (2008) *Being Human: Human-Computer Interaction in the Year 2020*, Microsoft Research Ltd.

HENRY, S. L. (2002) Understanding Web Accessibility. In THATCHER, J., BOHMAN, P., BURKS, M. R., HENRY, S. L., REGAN, B., SWIERENGA, S., URBAN, M. D. & WADDELL, C. D. (Eds.) *Constructing Accessibility Web Site,*. Birmingham, Glasshaus.

HENRY, S. L. (2007) Just Ask: Integrating Accessibility Throughout Design. Madison, WI, ET\Lawton.

HORNECKER, E. & BUUR, J. (2006) Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. *CHI 2006.* Montreal, Quebec, Canada.

HUBERMAN, B. A., ROMERO, D. M. & WU, F. (2008) *Social networks that matter: Twitter under the microscope*, arXiv:0812.1045v1 [cs.CY], accessed 4 Dec 2008.

HUGHES, J. (2001) *Sony AIBO*, http://the-gadgeteer.com/review/sony_aibo_review, accessed 22 January 2008.

HUTCHINSON, H., DRUIN, A. & BEDERSON, B. B. (2007) Designing Searching and Browsing Software for Elementary-Age Children. In LAZAR, J. (Ed.) *Universal Usability: Designing Computer Interfaces for Diverse Users.* West Sussex, John Wiley and Sons Ltd.

INKPEN, K. M. (1997) Three Important Research Agendas for Educational Multimedia: Learning, Children and Gender. *AACE World Conference on Educational Multimedia and Hypermedia 97.* Calgary.

ISYS (2000) *Making Information Usable: Interface Hall of Shame,* www.iarchitect.com/mshame.htm, accessed.

JOHNSON, C. (1997) *Interactive Systems Design*, http://www.dcs.gla.ac.uk/~johnson/teaching/isd/course.html, accessed 24 June 2010.

JONES, H. & SOLTREN, J. H. (2005) *Facebook: Threats to Privacy*, http://groups.csail.mit.edu/mac/classes/6.805/student-papers/fall05-papers/facebook.pdf, accessed 20 June 2010.

KAEMBA, M. (2008) *Digital Museum usability: A study with elderly users,* ftp://cs.joensuu.fi/pub/Theses/2008_MSc_Kaemba_Mitwa.pdf, accessed 18 April 2009.

KONSBRUCK, R. L. [sa]. *Impacts of Information Technology on Society in the new Century,* http://www.zurich.ibm.com/pdf/Konsbruck.pdf, accessed 1 July 2010.

KRAUS, L., STODDARD, S. & GILMARTIN, D. (2006) *Chartbook on Disability in the United States*, http://www.infouse.com/disabilitydata/disability/1_1.php, accessed 4 August 2009.

LEHOHLA, P. (2005) *Census 2001: Prevalence of Disability in South Africa*, http://www.statssa.gov.za/census01/html/Disability.pdf, accessed 22 July 2009.

LEPAGE, M., QIU, X. & SIFTON, V. (2006) *A New Role for Weather Simulations to Assist Wind Engineering Projects,* http://www.rwdi.com/cms/publications/47/t28.pdf, accessed 9 July 2010.

MERAKA INSTITUTE (Accessed 23 Oct 2007) *The Digital Doorway Project,* www.meraka.org.za/digitalDoorway.htm, accessed.

MIT (Accessed 23 Oct 2007) *One Laptop per Child Project,* www.laptopical.com/2b1-laptop.html, accessed.

MITRA, S. (2003) Minimally Invasive Education: A Progress Report on the "Hole-in-the-Wall" Experiments. *British Journal of Educational Technology,* 34**,** 367-371.

MONTEMAYOR, J., DRUIN, A. & HENDLER, J. (2000) PETS: A Personal Electronic Teller of Stories. In DRUIN, A. & HENDLER, J. (Eds.) *Robots for Kids: Exploring New Technologies for Learning.* Morgan Kaufman.

MUGLIA, B. (2010) *Modeling the World,* http://www.microsoft.com/mscorp/execmail/2010/05-17HPC.mspx, accessed 9 July 2010.

MYERS, B. A. (1998) A Brief History of Human Computer Interaction Technology. *ACM Interactions,* 5**,** 44-54.

NIELSEN, J. (1994) Heuristic Evaluation. In NIELSEN, J. & MACK, R. (Eds.) *Usability Inspection Methods.* New York, John Wiley and Sons.

NIELSEN, J. (2001) *Ten Usability Heuristics,* www.useit.com/papers/heuristic, accessed.

NISBETT, R. E. (2003) *The geography of thought: How Asians and Westerners think differently – and why.*, New York, Free Press.

NORMAN, D. A. (1999) *The Design of Everyday Things*, MIT Press.

PHIPPS, SUTHERLAND & SEALE. (2002) Access all areas: Disability, technology and learning. York, UK. TechDis with the Association for Learning Technology, 96pp

PLOWMAN, L. & STEPHEN, C. (2003) A 'benign addition'? Research on ICT and pre-schools children. *Journal of Computer Assisted Learning*, 19(2), 149-164

PREECE, J., ROGERS, Y. & SHARP, H. (2019) *Beyond human-computer interaction*. 5th Ed. Wiley, Sussex, London.

REASON, J. (1990) *Human error*, Cambridge University Press.

SMART, P.R. & Shadbolt, N.R. (2018) The World Wide Web. In J. Chase & D. Coady (Eds), Routledge *Handbook of Applied Epistemology*. Routledge, New York USA.

SHNEIDERMAN, B. (1998) *Designing the User Interface: Strategies for Effective Human-Computer Interaction,* New York, Addison-Wesley.

SHNEIDERMAN, PLASANT, COHEN & JACOBS. (2014). *Design the User Interface: Strategies for Effective Human-Computer Interaction*, Pearson Education Limited. Edinburg Gate, Essex, London.

SIMONITE, T. (2010) Online Translation Services Learn to Bridge the Language Gap. *New Scientist.*

STATISTICS SOUTH AFRICA (2001) *Census 2001: Key Results,* http://www.statssa.gov.za/census01/HTML/Key%20results_files/Key%20results.pdf, accessed 03 May 2010.

SWAMINATHAN, N. (2007) Could Robots Become Your Toddler's New Best Friend? *Scientific American.*

TANAKA, F., CIROUREL, A. & MOVELLEN, J. R. (2008) Socialization between toddlers and robots at an early childhood education center. National Academy of Science, ISSN 0027-92424, ISBN 19954-17958 DOI: https://doi.org/10.1073/pnas.0707769104, 104, 46

UNITED NATIONS (2006) *World Programme of Action Concerning Disabled Persons,* http://www.un.org/esa/socdev/enable/diswpa01.htm, accessed 30 September 2009.

UNIVERSITY OF MARYLAND HCI LAB (2008) *Project Screenshots,* www.cs.umd.edu/hcil/pubs/screenshots/PETS, accessed 22 January 2008.

WADDELL, C. D. (2002) Overview of Law and Guidelines. In THATCHER, J., BOHMAN, P., BURKS, M. R., HENRY, S. L., REGAN, B., SWIERENGA, S., URBAN, M. D. & WADDELL, C. D. (Eds.) *Constructing Accessible Web Sites.* Birmingham, Glasshaus.

WILLIGES, B. H. & WILLIGES, R. C. (1984) User Considerations in Computer-Based Information Systems. In MUCKLER, F. A. (Ed.) *Human Factors Review 1984.* Santa Monica, Human Factors Society.

WILSON, E. J. (2006) *The Information Revolution and Developing Countries,* Cambridge, MIT Press.

YBARRA, M. L. & MITCHELL, K. J. (2007) *How Risky Are Social Networking Sites? A Comparison of Places Online Where Youth Sexual Solicitation and Harassment Occurs,* www.pediatrics.org/cgi/doi/10.1542/peds.2007-0693, accessed 1 July 2010.

Young, Young and Lee Tolbert (2008) Clinical Neuroscience. ISBN 0781753198, 9780781753197. Lippincott Williams & Wilkins.