UNIT - I

PART - A

1.**Define reasoning. List its types**.(APR/MAY 2017)

Reasoning is the process by which we use the knowledge we have to draw conclusions or infer something new about domain of interest.

Types:

Deductive.
Inductive.
Abductive.

2.**What is ergnonomics?** .(APR/MAY 2017)

It is a Study of the physical characteristics of interaction. Also known as human factors but this can also be used to mean much of HCI. Ergonomics is good at defining standards and guidelines for constraining the way we design certain aspects of systems

Examples:

Arrangement of controls and displays
e.g. controls grouped according to function or frequency of use, or sequentially
surrounding environment
e.g. seating arrangements adaptable to cope with all sizes of user

3.Define Human computer interaction.

Human-computer interaction (HCI) is the study and planned design of human and computeractivities. HCI uses productivity, safety and entertainment to support and fulfill human-computeractivities and is applied to various types of computer systems, including air traffic control,nuclear processing, offices and computer gaming. HCI systems are easy, safe, effective andenjoyable.

4. What are the 5 major senses?

Sight
Hearing
Touch
Taste
Smell

5. What are the parts of the EyeCornea

Sclera
Conjunctiva
Iris
Pupil
Lens
Retina

6. What is meant by visual perception?

Visual perception is the ability to see, organize, and interpret one's environment. Visual perception is the ability to interpret the surrounding environment by processing information that is contained in visible light. The resulting perception is also known as eyesight, sight, or vision (adjectival form: visual, optical, or ocular). The various physiological components involved in vision are referred to collectively as the visual system, and are the focus of much research in linguistics, psychology, cognitive science, neuroscience, and molecular biology, collectively referred to as vision science.

7. What is sensory memory?

Buffers for stimuli received through senses
iconic memory: visual stimuli
echoic memory: aural stimuli
haptic memory:

8.What is long term memory? And mention its types

Long-term memory is intended for the long-term storage of information. There are two types of long-term memory: episodic memory and semantic memory.

Episodic memory:

It represents our memory of events and experiences in a serial form. It is from this memory that we can reconstruct the actual events that took place at a given point in our lives.

Semantic memory:

is a structured record of facts, concepts and skills that we have acquired. The information in semantic memory is derived from that in our episodic memory, such that we can learn new facts or concepts from our experiences.

9. Mention 7 stages of Donald norman's model in interaction?

user establishes the goal
formulates intention
specifies actions at interface
executes action
perceives system state
interprets system state
evaluates system state with respect to goal

10. List out common interaction style

command line interface
menus
natural language
question/answer and query dialogue
form-fills and spreadsheets
WIMP
point and click.

PART – B

1.Explain the model of the structure of human memory with diagrammatic illustration. .(APR/MAY 2017)

Sensory memory

The sensory memories act as buffers for stimuli received through the senses. A sensory memory exists for each sensory channel: *iconic memory* for visual stimuli, *echoic memory* for aural stimuli and *haptic memory* for touch. These memories are constantly overwritten by new information coming in on these channels.

Figure 1.9 A model of the structure of memory

Short-term memory

Short-term memory also has a limited capacity. There are two basic methods for measuring memory capacity. The first involves determining the length of a sequence which can be remembered in order. The second allows items to be freely recalled in any order. Using the first measure, the average person can remember 7 ± 2 digits. This was established in experiments by Miller [234]. Try it. Look at the following number sequence:
265397620853

Long-term memory

If short-term memory is our working memory or 'scratch-pad', long-term memory is our main resource. Here we store factual information, experiential knowledge, procedural rules of behavior – in fact, everything that we 'know'. It differs from short-term memory in a number of significant ways. First, it has a huge, if not unlimited, capacity. Secondly, it has a relatively slow access time of approximately a tenth of a second. Thirdly, forgetting occurs more slowly in long-term memory, if at all. These distinctions provide further evidence of a memory structure with several parts.

2.Outline the factors that can limit the speed of an interactive computer system.(APR/MAY 2017)

Limitations on interactive performance

There are several factors that can limit the speed of an interactive system:

Computation bound : This is rare for an interactive program, but possible, for example when using find/replace in a large document. The system should be designed so that long delays are not in the middle of interaction and so that the user gets some idea of how the job is progressing. For a very long process try to give an indication of duration *before* it starts; and during processing an indication of the stage that the process has reached is helpful. This can be achieved by having a counter or slowly filling bar on the screen that indicates the amount done, or by changing the cursor to indicate that processing is occurring. Many systems notice after they have been computing for some time and then say 'this may take
Some time: continue (Y/N)?'. Of course, by the time it says this the process may be nearly finished anyway!

Storage channel bound : As we discussed in the previous section, the speed of memory access can interfere with interactive performance. We discussed one technique, laziness, for reducing this effect. In addition, if there is plenty of raw computation power and the system is held up solely by memory, it is possible to trade off memory against processing speed. For example, compressed data take less space to store, and is faster to read in and out, but must be compressed before storage and decompressed when retrieved. Thus faster memory access leads to increased processing time. If data is written more often than it is read, one can choose a technique that is expensive to compress but fairly simple to decompress.

Graphics bound : For many modern interfaces, this is the most common bottleneck. It is easy to underestimate the time taken to perform what appear to be simple interface operations. Sometimes clever coding can reduce the time taken by common graphics operations, and there is tremendous variability in performance between programs running on the same hardware. Most computers include a special-purpose *graphics card* to handle many of the most common graphics operations.

Network capacity : Most computers are linked by networks. At the simplest this can mean using shared files on a remote machine. When accessing such files it can be the speed of the network rather than that of the memory which limits performance. This is discussed in greater detail below.

3) List and explain the stages of Norman's model of interaction. .(APR/MAY 2017)

The terms of interaction

Traditionally, the purpose of an interactive system is to aid a user in accomplishing *goals* from some application *domain*. (Later in this book we will look at alternative interactions but this model holds for many work-oriented applications.) A domain defines an area of expertise and knowledge in some real-world activity. Some examples of domains are graphic design, authoring and process control in a factory.

A domain consists of concepts that highlight its important aspects. In a graphic design domain, some of the important concepts are geometric shapes, a drawing surface and a drawing utensil. *Tasks* are operations to manipulate the concepts of a domain. A *goal* is the desired output from a performed task. For example, one task within the graphic design domain is the construction of a specific geometric shape with particular attributes on the drawing surface. A related goal would be to produce a solid red triangle centered on the canvas. An *intention* is a specific action required to meet the goal.

The execution–evaluation cycle

Norman's model of interaction is perhaps the most influential in Human–Computer Interaction, possibly because of its closeness to our intuitive understanding of the interaction between human user and computer [265]. The user formulates a plan of
action, which is then executed at the computer interface. When the plan, or part of the plan, has been executed, the user observes the computer interface to evaluate the result of the executed plan, and to determine further actions.

The interactive cycle can be divided into two major phases: execution and evaluation. These can then be subdivided into further stages, seven in all. The stages in Norman's model of interaction are as follows:

1. Establishing the goal.
2. Forming the intention.
3. Specifying the action sequence.
4. Executing the action.
5. Perceiving the system state.
6. Interpreting the system state.
7. Evaluating the system state with respect to the goals and intentions.

The interaction framework

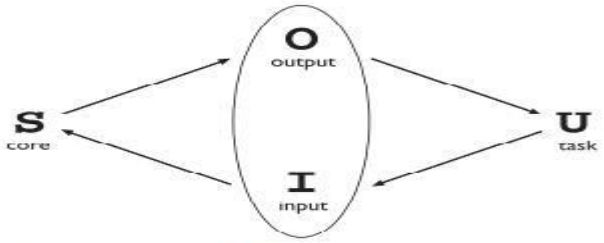


Figure 3.1 The general interaction framework

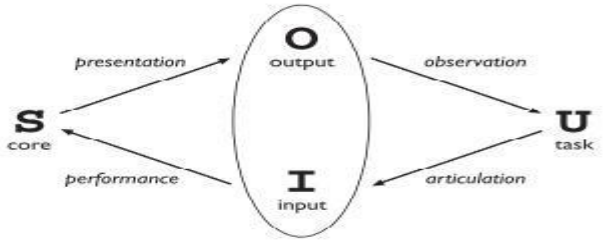


Figure 3.2 Translations between components

4.Outline the common interface styles used in interactive system.(APR/MAY 2017)

Interaction can be seen as a dialog between the computer and the user. The choice of interface style can have a profound effect on the nature of this dialog.. Here we introduce the most common interface styles and note the different effects these have on the interaction. There are a number of common interface styles including

command line interface
menus

natural language
question/answer and query dialog
form-fills and spreadsheets
WIMP
point and click
three-dimensional interfaces.

Command line interface

Menus

Ex
PAYMENT DETAILS
please select payment method:
1. cash
2. check
3. credit card
4. invoice
5. abort transaction

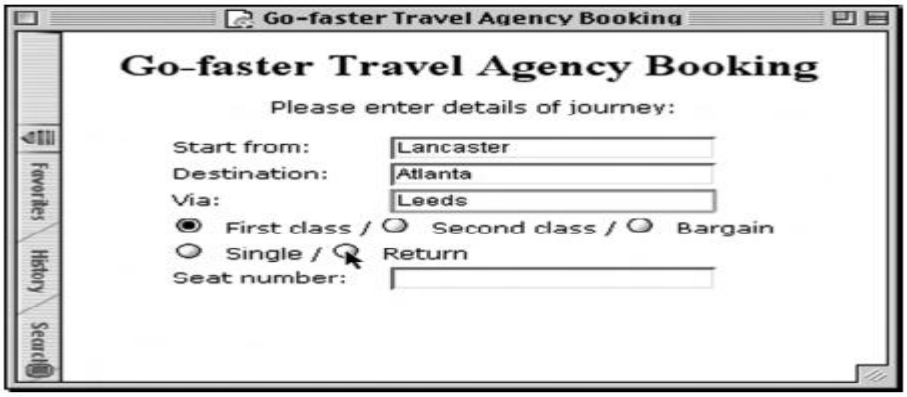
Natural language

The boy hit the dog with the stick

Question/answer and query dialog

Query languages, on the other hand, are used to construct queries to retrieve information from a database. They use natural-language-style phrases, but in fact require specific syntax, as well as knowledge of the database structure.

Form-fills and spreadsheets



The WIMP interface

WIMP stands for

windows, icons, menus and pointers (sometimes windows, icons, mice and pull-down menus), and is the default interface style for the majority of interactive computer systems in use today, especially in the PC and desktop workstation arena.

Point-and-click interfaces

5) Explain Input Output channels with examle

A person's interaction with the outside world occurs through information being received and sent: input and output.

Input in human is mainly though the senses and output through the motor control of the effectors. There are five major senses:

- Sight
- Hearing
- Touch
- Taste
- Smell

Vision

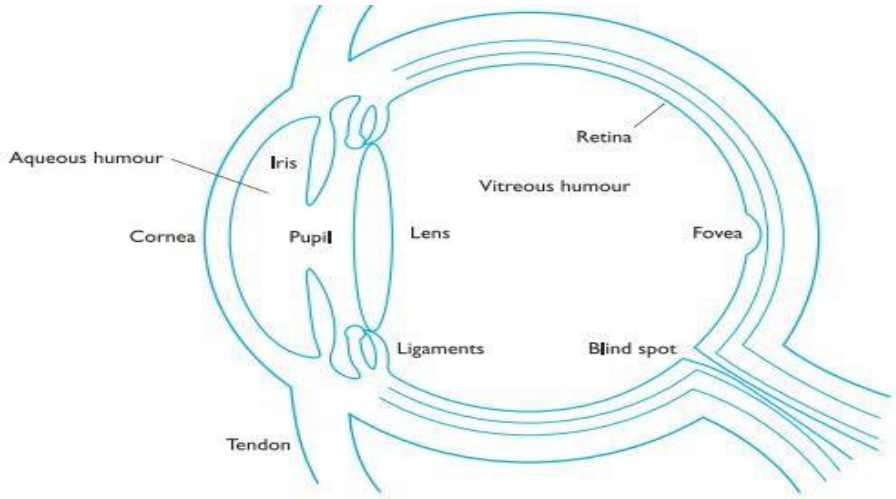


Figure 1.1 The human eye

The human eye

Vision begins with light. The eye is a mechanism for receiving light and transforming it into electrical energy. Light is reflected from objects in the world and their image is focused upside down on the back of the eye. The receptors in the eye transform it into electrical signals, which are passed to brain. The eye has a number of important components as you can see in the figure. Let us take a deeper look. The cornea and lens at the front of eye focus the light into a sharp image on the back of the eye, the retina. The retina is light sensitive and contains two types of photoreceptor: rods and cones.

Rods

Cones

Fovea

Blind spot

Nerve cells

The retina also has specialized nerve cells called ganglion cells.

X-cells

Y-cells

Visual perception

Perceiving size and depth

Visual acuity

Law of size constancy

Perceiving brightness

Hue

Intensity

Saturation

6. Explain Reasoning and problem solving.

We have considered how information finds its way into and out of the human system and how it is stored. Finally, we come to look at how it is processed and manipulated. This is perhaps the area which is most complex and which separates humans from other information-processing systems, both artificial and natural.

Although it is clear that animals receive and store information, there is little evidence to suggest that they can use it in quite the same way as humans. Similarly, artificial intelligence has produced machines which can see (albeit in a limited way) and store information. But their ability to use that information is limited to small domains.

Reasoning:

Reasoning is the process by which we use the knowledge we have to draw conclusions or infer something new about the domain of interest. There are a number of different types of reasoning: deductive, inductive and abductive. We use each of these types of reasoning in everyday life, but they differ in significant ways.

Deductive reasoning

Deductive reasoning derives the logically necessary conclusion from the given premises.

For example,

If it is Friday then she will go to work

It is Friday

Therefore she will go to work.

Inductive reasoning

Induction is generalizing from cases we have seen to infer information about cases we have not seen. For example, if every elephant we have ever seen has a trunk, we infer that all elephants

have trunks. Of course, this inference is unreliable and cannot be proved to be true; it can only be proved to be false. We can disprove the inference simply by producing an elephant without a trunk. However, we can never prove it true because, no matter how many elephants with trunks we have seen or are known to exist, the next one we see may be trunkless. The best that we can do is gather evidence to support our inductive inference.

Abductive reasoning

The third type of reasoning is abduction. Abduction reasons from a fact to the action or state that caused it. This is the method we use to derive explanations for the events we observe. For example, suppose we know that Sam always drives too fast when she has been drinking. If we see Sam driving too fast we may infer that she has been drinking. Of course, this too is unreliable since there may be another reason why she is driving fast: she may have been called to an emergency, for example.

7. Explain problem solving and its theory.

1.4.2 Problem solving

If reasoning is a means of inferring new information from what is already known, problem solving is the process of finding a solution to an unfamiliar task, using the knowledge we have. Human problem solving is characterized by the ability to adapt the information we have to deal with new situations. However, often solutions seem to be original and creative. There are a number of different views of how people solve problems. The earliest, dating back to the first half of the twentieth century, is the *Gestalt* view that problem solving involves both reuse of knowledge and insight. This has been largely superseded but the questions it was trying to address remain and its influence can be seen in later research. A second major theory, proposed in the 1970s by Newell and Simon, was the *problem space theory*, which takes the view that the mind is a limited information processor. Later variations on this drew on the earlier theory and attempted to reinterpret Gestalt theory in terms of informationprocessing theories. We will look briefly at each of these views.

Gestalt theory

Gestalt psychologists were answering the claim, made by behaviorists, that problem solving is a matter of reproducing known responses or trial and error. This explanation was considered by the Gestalt school to be insufficient to account for human problem-solving behavior. Instead, they claimed, problem solving is both *productive* and *reproductive*. Reproductive problem solving draws on previous experience as the behaviorists claimed, but productive problem solving involves insight and restructuring of the problem. Indeed, reproductive problem solving could be a hindrance to finding a solution, since a person may ‘fixate’ on the known aspects of the
Problem and so be unable to see novel interpretations that might lead to a solution. However, this was difficult to verify since the apes had once been wild and so could have been using previous knowledge.

Other experiments observed human problem-solving behavior. One well-known example of this is Maier’s *pendulum problem* [224]. The problem was this: the subjects were in a room with two pieces of string hanging from the ceiling. Also in the room were other objects including pliers, poles and extensions. The task set was to tie the pieces of string together. However, they

were too far apart to catch hold of both at once. Although various solutions were proposed by subjects, few chose to use the weight of the pliers as a pendulum to 'swing' the strings together. However, when the experimenter brushed against the string, setting it in motion, this
Solution presented itself to subjects. Maier interpreted this as an example of productive restructuring. The movement of the string had given insight and allowed the subjects to see the problem in a new way. The experiment also illustrates fixation: subjects were initially unable to see beyond their view of the role or use of a pair of pliers.

Problem space theory

Newell and Simon proposed that problem solving centers on the problem space. The problem space comprises *problem states*, and problem solving involves generating these states using legal state transition operators. The problem has an initial state and a goal state and people use the operators to move from the former to the latter. Such problem spaces may be huge, and so *heuristics* are employed to select appropriate operators to reach the goal. One such heuristic is *means–ends analysis*.

In means–ends analysis the initial state is compared with the goal state and an operator chosen to reduce the difference between the two. For example, imagine you are reorganizing your office and you want to move your desk from the north wall of the room to the window. Your initial state is that the desk is at the north wall. The goal state is that the desk is by the window. The main difference between these two is the location of your desk. You have a number of operators which you can apply to moving things: you can carry them or push them or drag them, etc. However, you know that to carry something it must be light and that your desk is heavy. You therefore have a new subgoal: to make the desk light. Your operators for this may involve removing drawers, and so on.

Analogy in problem solving

A third element of problem solving is the use of analogy. Here we are interested in how people solve novel problems. One suggestion is that this is done by mapping knowledge relating to a similar known domain to the new problem – called *analogical mapping*.

Similarities between the known domain and the new one are noted and operators from the known domain are transferred to the new one. This process has been investigated using

EX:

A doctor is treating a malignant tumor. In order to destroy it he needs to blast it with high-intensity rays. However, these will also destroy the healthy tissue sur- rounding the tumor. If he lessens the rays' intensity the tumor will remain. How does he destroy the tumor?]

The solution to this problem is to fire low-intensity rays from different directions converging on the tumor. That way, the healthy tissue receives harmless lowintensity rays while the tumor receives the rays combined, making a high-intensity dose. The investigators found that only 10% of subjects reached this solution without help. However, this rose to 80% when they were given this analogous story and told that it may help them:

Skill acquisition

All of the problem solving that we have considered so far has concentrated on handling unfamiliar problems. However, for much of the time, the problems that we face are not completely new. Instead, we gradually acquire skill in a particular domain area. But how is such skill acquired and what difference does it make to our problem-solving performance? We can

gain insight into how skilled behavior works, and how skills are acquired, by considering the difference between novice and expert behavior in given domains.
IF cook[type, ingredients, time]
THEN
cook for: time
cook[casserole, [chicken,carrots,potatoes], 2 hours]
cook[casserole, [beef,dumplings,carrots], 2 hours]
cook[cake, [flour,sugar,butter,eggs], 45 mins]
Gradually your knowledge becomes proceduralized and you have specific rules for

Errors and mental models

Human capability for interpreting and manipulating information is quite impressive. However, we do make mistakes. Some are trivial, resulting in no more than temporary inconvenience or annoyance. Others may be more serious, requiring substantial effort to correct. Occasionally an error may have catastrophic effects, as we see when 'human error' results in a plane crash or nuclear plant leak.

8. Explain in detail about Paradigms For Interaction.

4.2.1 Time sharing

One of the major contributions to come out of this new emphasis in research was the concept of *time sharing*, in which a single computer could support multiple users. Previously, the human (or more accurately, the programmer) was restricted to batch sessions, in which complete jobs were submitted on punched cards or paper tape to an operator who would then run them individually on the computer. Time-sharing systems of the 1960s made programming a truly interactive venture and brought about a subculture of programmers known as 'hackers' – single-minded masters of detail who took pleasure in understanding complexity. Though the purpose of the first interactive time-sharing systems was simply to augment the programming capabilities of the early hackers, it marked a significant stage in computer applications for human use.

Video display units

Sketchpad demonstrated two important ideas. First, computers could be used for more than just data processing. They could extend the user's ability to abstract away from some levels of detail, visualizing and manipulating different representations of the same information.

Programming toolkits

Building components of a computer system that will allow you to rebuild a more complex system is called bootstrapping and has been used to a great extent in all of computing. The power of programming toolkits is that small, well-understood components can be composed in fixed ways in order to create larger tools. Once these larger tools become understood, they can continue to be composed with other tools, and the process continues

Personal computing.

In the early 1970s his view of the future of computing was embodied in small, powerful machines which were dedicated to single users, that is *personal computers*. Together with the founding team of researchers at the Xerox Palo Alto Research Center (PARC), Kay worked on incorporating a powerful and simple visually based programming environment, Smalltalk, for the personal computing hardware that was just becoming feasible.

9. Explain Text entry devices in detail with an example.

The alphanumeric keyboard

The keyboard is still one of the most common input devices in use today. It is used for entering textual data and commands. The vast majority of keyboards have a standardized layout, and are known by the first six letters of the top row of alphabetical keys, QWERTY. There are alternative designs which have some advantages over the QWERTY layout, but these have not been able to overcome the vast technological inertia of the QWERTY keyboard. These lternatives are of two forms: 26 key layouts and chord keyboards. A 26 key layout rearranges the order of the alphabetic keys, putting the most commonly used letters under the strongest fingers, or adopting simpler practices. In addition to QWERTY, we will discuss two 26 key layouts, alphabetic and DVORAK, and chord keyboards.

The QWERTY keyboard

Figure 2.3 The standard QWERTY keyboard

Ease of learning – alphabetic keyboard

One of the most obvious layouts to be produced is the alphabetic keyboard, in which the letters are arranged alphabetically across the keyboard..

Chord keyboards

Chord keyboards are significantly different from normal alphanumeric keyboards. Only a few keys, four or five, and letters are produced by pressing one or more of the keys at once. For example, in the *Microwriter*, the pattern of multiple keypresses is chosen to reflect the actual letter shape.

2.2.3 Phone pad and T9 entry

Most phones have at least two *modes* for the numeric buttons: one where the keys mean the digits (for example when entering a phone number) and one where they mean letters (for example when typing an SMS message). Some have additional modes to make entering accented characters easier. Also a special mode or setting is needed for capital letters although many phones use rules to reduce this, for example automatically capitalizing the initial letter in a message and letters following full stops, question marks and exclamation marks.

10. Explain the types of display devices with example.

Bitmap displays – resolution and color

Virtually all computer displays are based on some sort of bitmap. That is the display is made of vast numbers of colored dots or pixels in a rectangular grid. These pixels may be limited to black and white (for example, the small display on many TV remote controls), in grayscale, or full color.

standard computer displays this is always in a 4:3 ratio, perhaps 1024 pixels across by 768 down, or 1600 × 1200; for PDAs this will be more in the order of a few hundred pixels in each direction. n the *density* of pixels: this is measured in pixels per inch.

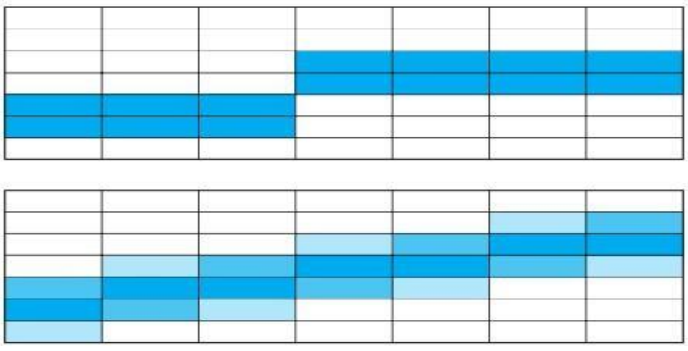


Figure 2.9 Magnified anti-aliased lines

Technologies

Cathode ray tube

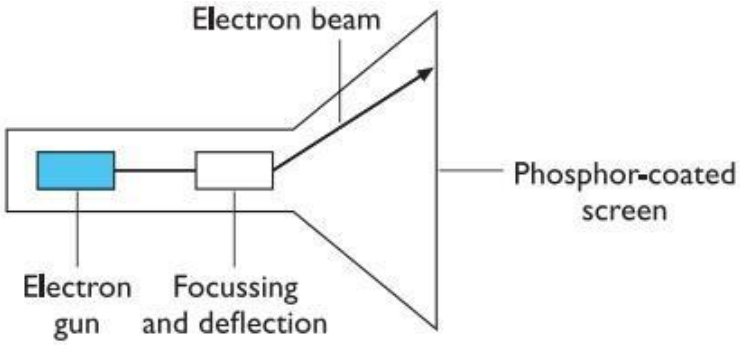


Figure 2.10 CRT screen

The CRT is a cheap display device and has fast enough response times for rapid animation coupled with a high color capability. Note that animation does not necessarily mean little creatures and figures running about on the screen, but refers in a more general sense to the use of motion in displays: moving the cursor, opening windows, indicating processor-intensive calculations, or whatever. As screen resolution increases, however, the price rises. Because of the electron gun and focussing components behind the screen, CRTs are fairly bulky, though recent innovations have led to flatter displays in which the electron gun is not placed so that it fires directly at the screen, but fires parallel to the screen plane with the resulting beam bent through 90 degrees to hit the screen.

Liquid crystal display

This different technology can be used to replace the standard screen on a desktop computer, and this is now common. However, the particular characteristics of compactness, light weight and low power consumption have meant that these screens have created a large niche in the computer market by monopolizing the notebook and portable computer systems side. The advent of these screens allowed small, light computers to be built, and created a large market that did not previously exist. Such computers, riding on the back of the technological wave, have opened up a different way of working for many people, who now have access to computers when away from the office, whether out on business or at home. Working in a different location on a smaller machine with different software obviously represents a different style.

Digital paper

A new form of ‘display’ that is still in its infancy is the various forms of digital paper. These are thin flexible materials that can be written to electronically, just like a computer screen, but which keep their contents even when removed from any electrical supply.

UNIT- II
PART - A

1. **Define prototyping**. (APR/MAY 2017)

   An iterative design is described by the use of prototypes.

   An iterative design process tries to overcome inherent problems of incomplete requirement specification by cycling through several designs, incrementally improving upon the final product with each pass.

2. **What do you mean by universal design?** .(APR/MAY 2017)

   Universal design is the process of designing interactive systems that are usable by anyone, with any range abilities, using any technology platform.
   This can be achieved by designing systems either to have built in redundancy or to be compatible with assistive technologies.

3. **What is usability engineering?**

   Usability engineering implies more of a focus on assessing and making recommendations to improve usability than it does on design. Usability Engineers may still engage in design to some extent, particularly through the design of wire-frames or other prototypes.

4. **Define model of software life cycle.**

   SDLC, Software Development Life Cycle is a process used by software industry to design, develop and test high quality softwares. The SDLC aims to produce a high quality software that meets or exceeds customer expectations, reaches completion within times and cost estimate Stage
   1: Planning and Requirement Analysis Stage
   2: Defining Requirements Stage
   3: Designing the product architecture Stage
   4: Building or Developing the Product Stage
   5: Testing the Product Stage
   6: Deployment in the Market and Maintenance

5. **Define cognitive walkthrough**

   Cognitive walkthrough is task-specific, whereas heuristic evaluation takes a holistic view to catch problems not caught by this and other usability inspection methods. The method is rooted in the notion that users typically prefer to learn a system by using it to accomplish tasks, rather than, for example, studying a manual. The method is prized for its ability to generate results quickly with low cost, especially when compared to usability testing, as well as the ability to apply the method early in the design phases, before coding even begins.

6. **What is Heuristic Evaluation**

   A heuristic evaluation is a usability inspection method for computer software that helps to identify usability problems in the user interface (UI) design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the"heuristics"). These evaluation methods are now widely taught and practiced in the new media sector, where UIs are often

designed in a short space of time on a budget that may restrict the amount of money available to provide for other types of interface testing.

7. **List different styles of Evaluation**

Inspection methods (no users needed!)
Heuristic evaluations
Walkthroughs
Other Inspections
User Tests (users needed!)
Observations/Ethnography
Usability tests/ Controlled Experiments

8. **What is think aloud**

Think aloud is a form of observation where the user is asked to talk through what he is doing as he is being observed; Think aloud has the advantage of simplicity; it requires little expertise to perform (though can be tricky to analyze fully) and can provide useful insight into problems with an interface. It can also be employed to observe how the system is actually used. It can be used for evaluation throughout the design process, using paper or simulated mock-ups for the earlier stages.

9. **Mention about query techniques**

Query techniques can be useful in eliciting detail of the user‟s view of a system. They embody the philosophy that states that the best way to find out how a system meets user requirements is to „ask the user‟. They can be used in evaluation and more widely to collect information about user requirements and tasks. The advantage of such methods is that they get the user‟s viewpoint directly and may reveal issues that have not been considered by the designer.

10. **Mention the process of design**

A system has been designed and built, and only when it proves unusable do they think to ask how to do it right! In other companies usability is seen as equivalent to testing checking whether people can use it and fixing problems, rather than making sure they can from the beginning.

Identifying needs and establishing requirements
Developing alternative designs that meet those requirements
Building interactive versions of the designs so that they can be communicated and assessed

PART - B

1. **With a neat sketch ,explain the interaction design process in. (APR/MAY 2017)**

So what is design? A simple definition is:Achieving goals within constraints This does not capture everything about design, but helps to focus us on certain things:

Goals What is the purpose of the design we are intending to produce? Who is it for? Why do they want it? For example, if we are designing a wireless personal movie player, we may think

about young affluent users wanting to watch the latest movies whilst on the move anddownload free copies, and perhaps wanting to share the experience with a few friends.

To err is human

It might sound demeaning to regard people as 'materials', possibly even dehumanizing. In fact, the opposite is the case: physical materials are treated better in most designs than people. This is particularly obvious when it comes to failures. The news headlines: an aircrash claims a hundred lives; an industrial accident causes millions of pounds' worth of damage; the discovery of systematic mistreatment leads to thousands of patients being recalled to hospital. Some months later the public inquiries conclude: human error in the operation of technical instruments.

THE PROCESS OF DESIGN

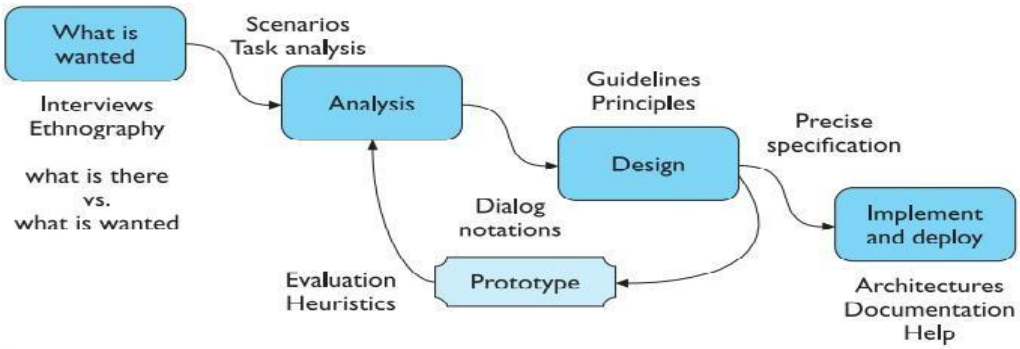


Figure 5.1 Interaction design process

There are a number of techniques used for this in HCI: interviewing people, videotaping them, looking at the documents and objects that they work with, observing them directly.

Analysis The results of observation and interview need to be ordered in some way to bring out key issues and communicate with later stages of design.

Design Well, this is all about design, but there is a central stage when you move from what you want, to how to do it.

Iteration and prototyping Humans are complex and we cannot expect to get designs right first time. We therefore need to evaluate a design to see how well it is working and where there can be improvements.

Implementation and deployment Finally, when we are happy with our design, we need to create it and deploy it. This will involve writing code, perhaps making hardware, writing documentation and manuals.

2. Narrate the shneiderman's eight golden rules of interface design. .(APR/MAY 2017)

1. Strive for consistency in action sequences, layout, terminology, command use and so on.

2. Enable frequent users to use shortcuts, such as abbreviations, special key sequences and macros, to perform regular, familiar actions more quickly.

3. Offer informative feedback for every user action, at a level appropriate to the magnitude of the action.

4. Design dialogs to yield closure so that the user knows when they have completed a task.

5. Offer error prevention and simple error handling so that, ideally, users are prevented from making mistakes and, if they do, they are offered clear and informative instructions to enable them to recover.

6. Permit easy reversal of actions in order to relieve anxiety and encourage exploration, since the user knows that he can always return to the previous state.

7. Support internal locus of control so that the user is in control of the system, which responds to his actions.

8. Reduce short-term memory load by keeping displays simple, consolidating multiple page displays and providing time for learning action sequences.

3. Outline the approaches used for evaluation through expert analysis.(APR/MAY 2017)

Cognitive walkthrough

The origin of the cognitive walkthrough approach to evaluation is the code walkthrough familiar in software engineering. Walkthroughs require a detailed review of a sequence of actions. In the code walkthrough, the sequence represents a segment of the program code that is stepped through by the reviewers to check certain characteristics (for example, that coding style is adhered to, conventions for spelling variables versus procedure calls, and to check that system-wide invariants are not violated). In the cognitive walkthrough, the sequence of actions refers to the steps that an interface will require a user to perform in order to accomplish some known task. The evaluators then 'step through' that action sequence to check it for potential usability problems.

A specification or prototype of the system. It doesn't have to be complete, but it should be fairly detailed. Details such as the location and wording for a menu can make a big difference.

A description of the task the user is to perform on the system. This should be a representative task that most users will want to do.
A complete, written list of the actions needed to complete the task with the proposed system.
An indication of who the users are and what kind of experience and knowledge the evaluators can assume about them.

Heuristic evaluation

A heuristic is a guideline or general principle or rule of thumb that can guide a design decision or be used to critique a decision that has already been made. Heuristic evaluation, developed by Jakob Nielsen and Rolf Molich, is a method for structuring the critique of a system using a set of relatively simple and general heuristics. Heuristic evaluation can be performed on a design specification so it is useful for evaluating early design. But it can also be used on prototypes, storyboards and fully functioning systems. It is therefore a flexible, relatively cheap approach. Hence it is often considered a discount usability technique.

Model-based evaluation

A third expert-based approach is the use of models. Certain cognitive and design models provide a means of combining design specification and evaluation into the same framework. These are discussed in detail in Chapter 12. For example, the GOMS (goals, operators, methods and selection) model predicts user performance with a particular interface and can be used to filter particular design options. Similarly, lower-level modeling techniques such as the keystroke-level model provide predictions of the time users will take to perform low-level physical tasks.

4.Explain the approaches to evaluate expert through user participation.

Styles of evaluation

Laboratory studies

In the first type of evaluation studies, users are taken out of their normal work environment to take part in controlled tests, often in a specialist usability laboratory (although the ‘lab’ may simply be a quiet room). This approach has a number of benefits and disadvantages. A well-equipped usability laboratory may contain sophisticated audio/visual recording and analysis facilities, two-way mirrors, instrumented computers and the like, which cannot be replicated in the work environment. In addition, the participant operates in an interruption-free environment.

Field studies

The second type of evaluation takes the designer or evaluator out into the user's work environment in order to observe the system in action. Again this approach has its pros and cons.

Empirical methods: experimental evaluation

Any experiment has the same basic form. The evaluator chooses a hypothesis to test, which can be determined by measuring some attribute of participant behavior. A number of experimental conditions are considered which differ only in the values of certain controlled variables. Any changes in the behavioral measures are attributed to the different conditions. Within this basic form there are a number of factors that are important to the overall reliability of the experiment, which must be considered carefully in experimental design.

Participants

The choice of participants is vital to the success of any experiment. In evaluation experiments, participants should be chosen to match the expected user population as closely as possible. Ideally, this will involve experimental testing with the actual users but this is not always possible. If participants are not actual users, they should be chosen to be of a similar age and level of education as the intended user group.

Variables

Hypotheses

A hypothesis is a prediction of the outcome of an experiment. It is framed in terms of the independent and dependent variables, stating that a variation in the independent variable will cause a difference in the dependent variable. The aim of the experiment is to show that this prediction is correct.

Experimental design

The next step is to decide on the *experimental method* that you will use. There are two main methods: *between-subjects* and *within-subjects*. In a between-subjects (or *randomized*) design, each participant is assigned to a different condition. There are at least two conditions: the experimental condition (in which the variable has been manipulated) and the control, which is identical to the experimental condition except for this manipulation.

Statistical measures

Variables can be classified as either *discrete variables* or *continuous variables*. A discrete variable can only take a finite number of values or *levels*, for example, a screen color that can be red, green or blue. A continuous variable can take any value (although it may have an upper or lower limit), for example a person's height or the time taken to complete a task. A special case of continuous data is when they are *positive*.

Observational techniques

A popular way to gather information about actual use of a system is to observe users interacting with it. Usually they are asked to complete a set of predetermined tasks, although if observation is being carried out in their place of work, they may be observed going about their normal duties.

Protocol analysis

Paper and pencil Audio recording

Video recording

Computer logging

User notebooks

5.Explain HCI Patterns.

A pattern is an invariant solution to a recurrent problem within a specific context. Patterns address the problems that designers face by providing a 'solution statement'. This is best illustrated by example. Alexander, who initiated the pattern concept, proposes a pattern for house building called 'Light on Two Sides of Every Room'. The problem being addressed here is that when they have a choice, people will always gravitate to those rooms which have light on two sides, and leave the rooms which are lit only from one side unused and empty.

Note that the solution says nothing about where these windows should be located or at what angle they should be to each other. A room with windows on opposite walls, or at right angles, or with a window and a skylight would all fulfill the pattern. Patterns capture only the invariant properties of good design – the common elements that hold between all instances of the solution. The specific implementation of the pattern will depend on the circumstance and the designer's creativity.

Patterns and pattern languages are characterized by a number of features, which, taken as a whole, distinguish them from other design rules:

They capture design practice and embody knowledge about successful solutions: they come from practice rather than psychological theory.

They capture the essential common properties of good design: they do not tell the designer *how* to do something but what needs to be done and why.

They represent design knowledge at varying levels, ranging from social and organizational issues through conceptual design to detailed widget design.

They are not neutral but embody values within their rationale. Alexander's language clearly expresses his values about architecture. HCI patterns can express values about what is humane in interface design.

The concept of a pattern language is generative and can therefore assist in the development of complete designs.

They are generally intuitive and readable and can therefore be used for communication between all stakeholders.

6. What are the standards followed for designing process.Explain in detail.

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 specifically to either the hardware or the software used to build the interactive system.

Underlying theory Standards for hardware are based on an understanding of physiology or ergonomics/human factors, the results of which are relatively well known, fixed and readily adaptable to design of the hardware. On the other hand, software standards are based on theories from psychology or cognitive science, which are less well formed, still evolving and not very easy to interpret in the language of software design.

Change Hardware is more difficult and expensive to change than software, which is usually designed to be very flexible. Consequently, requirements changes for hardware do not occur as frequently as for software. Since standards are also relatively stable, they are more suitable for hardware than software.

One component of the ISO standard 9241, pertainin g to usability specification,applies equally to both hardware and software design. In the beginning of that document, the following definition of usability is given: **Usability**

The effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments.

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.

7. What are the principles followed to support usability?Explain in detail.

Learnability – the ease with which new users can begin effective interaction and achieve maximal performance.
Flexibility – the multiplicity of ways in which the user and system exchange information.
Robustness – the level of support provided to the user in determining successful achievement and assessment of goals.
Learnability

Principle	Definition	Related principles
Predictability	Support for the user to determine the effect of future action based on past interaction history	Operation visibility
Synthesizability	Support for the user to assess the effect of past operations on the current state	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	Guessability, affordance
Generalizability	Support for the user to extend knowledge of specific interaction within and across applications to other similar situations	–
Consistency	Likeness in input–output behavior arising from similar situations or similar task objectives	–

Flexibility

Dialog initiative	Allowing the user freedom from artificial constraints on the input dialog 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, multi-modality
Task migratability	The ability to pass control for the execution of a given task so that it becomes either internalized by the user or the system or shared between them	–
Substitutivity	Allowing equivalent values of input and output to be arbitrarily substituted for each other	Representation multiplicity, equal opportunity
Customizability	Modifiability of the user interface by the user or the system	Adaptivity, adaptability

Robustness

Principle	Definition	Related principles
Observability	Ability of the user to evaluate the internal state of the system from its perceivable representation	Browsability, static/dynamic defaults, reachability, persistence, operation visibility
Recoverability	Ability of the user to take corrective action once an error has been recognized	Reachability, forward/ backward recovery, commensurate effort
Responsiveness	How the user perceives the rate of communication with the system	Stability
Task conformance	The degree to which the system services support all of the tasks the user wishes to perform and in the way that the user understands them	Task completeness, task adequacy

8. Explain design rationale with example.

In designing any computer system, many decisions are made as the product goes from a set of vague customer requirements to a deliverable entity. Often it is difficult to recreate the reasons, or rationale, behind various design decisions.

Design rationale is the information that explains why a computer system is the way it is, including its structural or architectural description and its functional or behavioral description. In this sense, design rationale does not fit squarely into the software life cycle described in this chapter as just another phase or box. Rather, design rationale relates to an activity of both reflection (doing design rationale) and documentation (creating a design rationale) that occurs throughout the entire life cycle.

It is beneficial to have access to the design rationale for several reasons:

In an explicit form, a design rationale provides a communication mechanism among the members of a design team so that during later stages of design and/or maintenance it is possible to understand what critical decisions were made, what alternatives were investigated (and, possibly, in what order) and the reason why one alternative was chosen over the others. This can help avoid incorrect assumptions later.

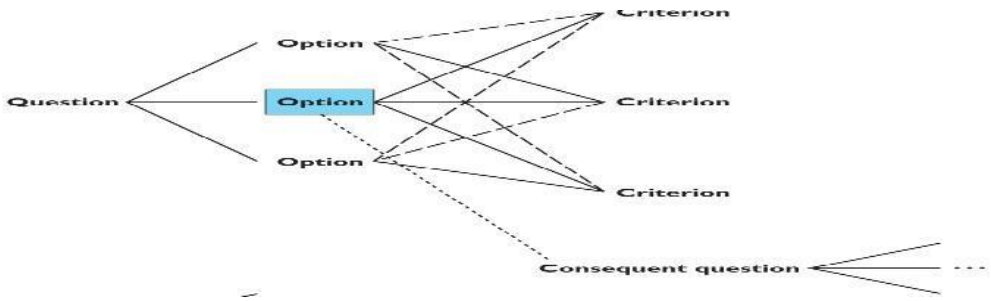
Accumulated knowledge in the form of design rationales for a set of products can be reused to transfer what has worked in one situation to another situation which has similar needs. The design rationale can capture the context of a design decision in order that a different design team can determine if a similar rationale is appropriate for their product.

The effort required to produce a design rationale forces the designer to deliberate more carefully about design decisions. The process of deliberation can be assisted by the design rationale technique by suggesting how arguments justifying or discarding a particular design option are formed.

Process-oriented design rationale

In IBIS (pronounced ‘ibbiss’), a hierarchical structure to a design rationale is created. A root issue is identified which represents the main problem or question that the argument is addressing. Various positions are put forth as potential resolutions for the root issue, and these are depicted as descendants in the IBIS hierarchy directly connected to the root issue. Each position is then supported or refuted by arguments, which modify the relationship between issue and position. The hierarchy grows as secondary issues are raised which modify the root issue in some way. Each of these secondary issues is in turn expanded by positions and arguments, further sub-issues, and so on.

Design space analysis



Psychological design rationale

People use computers to accomplish some tasks in their particular work domain, a s we have seen before. When designing a new interactive system, the designers take into account the tasks that users currently perform and any new ones that they may want to perform. This task identification serves as part of the requirements for the new system, and can be done through empirical observation of how people perform their work currently and presented through informal language or a more formal task analysis language.

9. **Explain iterative design and prototyping**

On the technical side, iterative design is described by the use of prototypes, artifacts that simulate or animate some but not all features of the intended system. There are three main approaches to prototyping:

Throw-away The prototype is built and tested. The design knowledge gained from this exercise is used to build the final product, but the actual prototype is discarded. Figure 6.5 depicts the procedure in using throw-away prototypes to arrive at a final requirements specification in order for the rest of the design process to proceed.

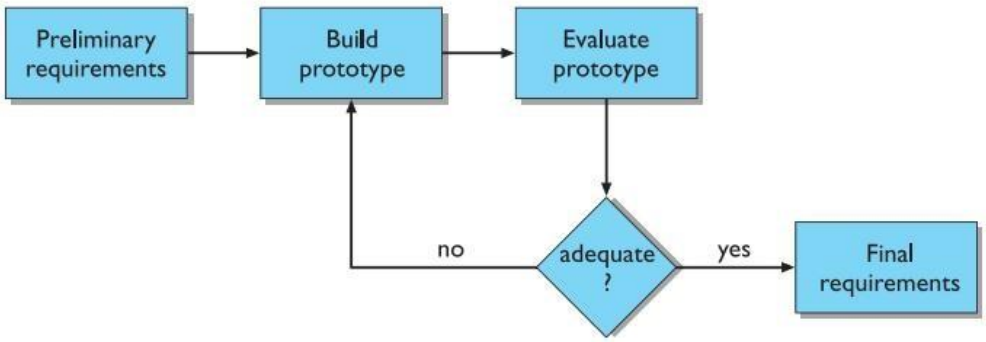


Figure 6.5 Throw-away prototyping within requirements specification

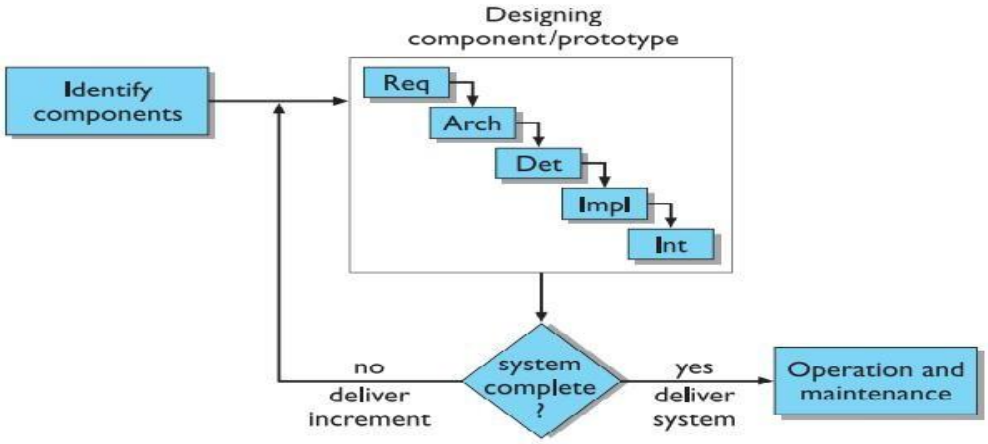


Figure 6.6 Incremental prototyping within the life cycle

Incremental The final product is built as separate components, one at a time. There is one overall design for the final system, but it is partitioned into independent and smaller compone nts. The final product is then released as a series of products, each subsequent release including one more component.

This is **Evolutionary** Here the prototype is not discarded and serves as the basis for the next iteration of design. Evolutionary prototyping also fits in well with the modifications which must be made to the system that arise during the operation and maintenance activity in the life cycle.

10. Explain usability engineering.

Attribute:	Backward recoverability
Measuring concept:	Undo an erroneous programming sequence
Measuring method:	Number of explicit user actions to undo current program
Now level:	No current product allows such an undo
Worst case:	As many actions as it takes to program in mistake
Planned level:	A maximum of two explicit user actions
Best case:	One explicit cancel action

The backward recoverability attribute is defined in terms of a measuring concept, which makes the abstract attribute more concrete by describing it in terms of the actual product. So in this case, we realize backward recoverability as the ability to undo an erroneous programming sequence. The measuring method states how the attribute will be measured, in this case by the number of explicit user actions required to perform the undo, regardless of where the user is in the programming sequence.

Set levels with respect to information on:
1. an existing system or previous version
2. competitive systems
3. carrying out the task without use of a computer system
4. an absolute scale
5. your own prototype
6. user's own earlier performance
7. each component of a system separately
8. a successive split of the difference between best and worst values observed in user Tests

Table 6.4 Examples of usability metrics from ISO 9241

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Suitability for the task	Percentage of goals achieved	Time to complete a task	Rating scale for satisfaction
Appropriate for trained users	Number of power features used	Relative efficiency compared with an expert user	Rating scale for satisfaction with power features
Learnability	Percentage of functions learned	Time to learn criterion	Rating scale for ease of learning
Error tolerance	Percentage of errors corrected successfully	Time spent on correcting errors	Rating scale for error handling

Problems with usability engineering

The major feature of usability engineering is the assertion of explicit usability metrics early on in the design process which can be used to judge a system once it is delivered. There is a very solid argument which points out that it is only through empirical approaches such as the use of usability metrics that we can reliably build more usable systems. Although the ultimate yardstick for determining usability may be by observing and measuring user performance, that does not mean that these measurements are the best way to produce a predictive design process for usability.

UNIT – III

PART - A

1. **List out the types of textual communication**. .(APR/MAY 2017)

   There are four types of textual communication in current groupware:

   1. **discrete** – directed message as in email. There is no explicit connection between different messages, except in so far as the text of the message refers to a previous one.
   2. **linear** – participants' messages are added in (usually temporal) order to the end of a single transcript.
   3. **non-linear** – when messages are linked to one another in a hypertext fashion.
   4. **spatial** – where messages are arranged on a two-dimensional surface.

2. **What is meant by GOMS? Give an example.Goals**

   GOMS are symbolic structures that define a state of affairs to be achieved and determinate a set of possible methods by which it may be accomplished

   Operators

   Operators are elementary perceptual, motor or cognitive acts, whose execution is necessary to change any aspect of the user's mental state or to affect the task environment

   Methods

   It describe a procedure for accomplishing a goal

   Selection

   Rules are needed when a goal is attempted; there may be more than onemethod available to the user to accomplish it.

3. **What is cognitive model?**

   Cognitive modeling deals with simulating human problem solving and mental task processes in a computerized model. Cognitive modeling is used in numerous artificial intelligence (AI) applications, such as expert systems, natural language programming, and neural networks, and in robotics and virtual reality application .Cognitive models are also used to improve products in manufacturing segments such as human factors engineering, and computer game and user interface design. Research into cognitive modeling is currently being conducted by academic and industry groups, including MIT, IBM, and Sandia National Laboratories.

4. **What is meant by Backups Naur Form? Give an example.**

   Backus Naur form (BNF) is a notation technique for context-free grammars, often used to describe the syntax of languages used in computing, such as computer programming languages, document formats, instruction sets and communication protocols. They are applied wherever exact descriptions of languages are needed: for instance, in official language specifications, in manuals, and in textbooks on programming language theory.

   draw-line ::= select-line + choose-points + last-point select-line ::= position-mouse + CLICK-MOUSE choose-points ::= choose-one| choose-one + choose-points choose-one ::= position-mouse + CLICK-MOUSE last-point ::= position-mouse + DOUBLE-CLICK-MOUSE position-mouse ::= empty | MOVE-MOUSE + position-mouse

5. **What is Task Action Grammar?**
   Task action grammar (TAG) attempts to deal with some of these problems by including elements such as parameterized grammar rules to emphasize consistency and encoding the user's world knowledge (for example, up is the opposite of down).
6. **What is Key stroke Level Model?**
   Keystroke-level model (KLM) predicts how long it will take an expert user to accomplish a routine task without errors using an interactive computer system.

7. **What is meant by problem space model?**
   A problem space consists of a set of states and a set of operations that can be performed on the states. Behavior in a problem space is a two-step process. The current operator is chosen based on the current state and then it is applied to the current state to achieve the new state. The problem space must represent rational behavior, and so it must characterize the goal of the agent. A problem space represents a goal by defining the desired states as a subset of all possible states. Once the initial state is set, the task within the problem space is to find a sequence of operations that form a path within the state space from the initial state to one of the desired states, whereupon successful termination occurs.

8. **Who are stakeholders? What roles do they play?**
   A person, group or organization that has interest or concern in an organization.Stakeholders can affect or be affected by the organization's actions, objectives and policies.Some examples of key stakeholders are creditors, directors, employees, government (andits agencies), owners(shareholders), suppliers, unions, and the community from whichthe business draws its resources. Not all stakeholders are equal. A company's customersare entitled to fair trading practices but they are not entitled to the same consideration asthe company's employees.An example of a negative impact on stakeholders is when a company needs to cut costsand plans a round of layoffs.

9. **What is static and dynamic web content?**
   Static Web pages display the exact same information whenever anyone visits it. StaticWeb pages do not have to be simple plain text. They can feature detailed multimediadesign and even videos. However, every visitor to that page will be greeted by the exactsame text, multimedia design or video every time he visits the page until you alter that page's source code.

10. Highlight the applications of hypermedia. APR/MAY 2017

• rapid prototyping
create live storyboards
mock-up interaction using links
help and documentation
allows hierarchical contents, keyword search or browsing
just in time learning
links between similar photocopiers
education animation and graphics allow students to see things happen sound adds atmosphere and means diagrams can be looked at while hearing explanation e-learning

PART - B

1.What is cognitive model? Classify cognitive models and discuss the same.(APR/MAY 2017)

Categories:
hierarchical representation of the user's task and goal structure
linguistic and grammatical models
physical and device-level models

GOAL AND TASK HIERARCHIES
Many models make use of a model of mental processing in which the user achieves goals by solving subgoals in a divide-and-conquer fashion. We will consider two models, *GOMS* and *CCT*, where this is a central feature.

produce report
gather data
. find book names
. . do keywords search of names database
<<further subgoals>>
. . sift through names and abstracts by hand
<<further subgoals>>
. search sales database
<<further subgoals>>
layout tables and histograms
<<further subgoals>>
write description
<<further subgoals>>

GOMS

Goals These are the user's goals, describing what the user wants to achieve. Further, in GOMS the goals are taken to represent a 'memory point' for the user, from which he can evaluate what should be done and to which he may return should any errors occur.

Operators These are the lowest level of analysis. They are the basic actions that the user must perform in order to use the system. They may affect the system (for example, press the 'X' key) or only the user's mental state (for example, read the dialog box). There is still a degree of flexibility about the granularity of operators; we may take the command level 'issue the SELECT command' or be more primitive: 'move mouse to menu bar, press center mouse button...'.

Methods As we have already noted, there are typically several ways in which a goal can be split into subgoals. For instance, in a certain window manager a currently selected window can be closed to an icon either by selecting the 'CLOSE' option from a pop-up menu, or by hitting the

'L7' function key. In GOMS these two goal decompositions are referred to as methods, so we have the CLOSE-METHOD and
the L7-METHOD:

Selection From the above snippet we see the use of the word select where the choice of methods arises. GOMS does not leave this as a random choice, but attempts to predict which methods will be used. This typically depends both on the particular user and on the state of the system and details about the goals. For instance, a user, Sam, never uses the L7-METHOD, except for one game, 'blocks', where the mouse needs to be used in the game until the very moment
the key is pressed. GOMS captures this in a selection rule for Sam:

User Sam:
Rule 1: Use the CLOSE-METHOD unless another rule applies.
Rule 2: If the application is 'blocks' use the L7-METHOD.

Cognitive complexity theory

The production rules are a sequence of rules:
if *condition* then *action*

Problems and extensions of goal hierarchies

GOAL: GET-MONEY
. GOAL: USE-ATM
. . INSERT-CARD
. . ENTER-PIN
. . ENTER-AMOUNT
. . COLLECT-MONEY
<< outer goal now satisfied goal stack popped >>
. . COLLECT-CARD – subgoal operators missed

2. **who is stakeholders?outline the types of stakeholders and appraise the stakeholders for an airline booking system.(APR/MAY 2017)**

Understanding stakeholders is key to many of the approaches to requirements capture, since in an organizational setting it is not simply the end-user who is affected by the introduction of new technology. Imagine that a new billing system is to be introduced by a local gas supplier. Who will be affected by this decision? Obviously, the people who are responsible for producing and sending out bills – they will be the ones using the system directly.

Primary stakeholders are people who actually use the system – the end-users.

Secondary stakeholders are people who do not directly use the system, but receive output from it or provide input to it (for example, someone who receives a report produced by the system).

Tertiary stakeholders are people who do not fall into either of the first two categories but who are directly affected by the success or failure of the system (for example,a director whose profits increase or decrease depending on the success of the system).

Facilitating stakeholders are people who are involved with the design, development and maintenance of the system.

Classifying stakeholders – an airline booking system

An international airline is considering introducing a new booking system for use by associated travel agents to sell flights directly to the public. The stakeholders can be classified as follows:

Primary stakeholders: travel agency staff, airline booking staff

Secondary stakeholders: customers, airline management

Tertiary stakeholders: competitors, civil aviation authorities, customers' traveling companions,airline shareholders

Facilitating stakeholders: design team, IT department staff

The aim of the design team is to meet the needs of as many stakeholders as possible. However, the reality is that usually stakeholder needs are in conflict with each other.Sometimes this does not matter: a company is unlikely to be too concerned that its competitors' requirement to maintain advantage over it is under threat by the new system (though they need to be aware to monitor how effectively they are maintaining their advantage). However, sometimes it does matter. In the example above, the airline booking system must be usable by travel agency staff but must also fulfill the customer need to find an appropriate ticket at the right price. If it fails in this, the whole system will fail, as the customer will go elsewhere and business will be lost.

3. **Explain the Socio-technical models in detail.**

Socio-technical models for interactive systems are therefore concerned with technical, social, organizational and human aspects of design. They recognize the fact that technology is not developed in isolation but as part of a wider organizational environment. It is important to consider social and technical issues side by side so that human issues are not overruled by technical considerations.

The problem being addressed: there is a need to understand why the technology is being proposed and what problem it is intended to solve.

The stakeholders affected, including primary, secondary, tertiary and facilitating, together with their objectives, goals and tasks.

The workgroups within the organization, both formal and informal.

The changes or transformations that will be supported.
The proposed technology and how it will work within the organization.
External constraints and influences and performance measures.

CUSTOM methodology

CUSTOM is a socio-technical methodology designed to be practical to use in small organizations [200]. It is based on the User Skills and Task Match (USTM) approach, developed to allow design teams to understand and fully document user requirements [219]. CUSTOM focuses on establishing stakeholder requirements:

all stakeholders are considered, not just the end-users.

1. Describe the organizational context, including its primary goals, physical characteristics, political and economic background.
2. Identify and describe stakeholders. All stakeholders are named, categorized (as primary, secondary, tertiary or facilitating) and described with regard to personal issues, their role in the organization and their job. For example, CUSTOM addresses issues such as stakeholder motivation, disincentives, knowledge, skills, power and influence within the organization, daily tasks and so on.
3. Identify and describe work-groups. A work-group is any group of people who work together on a task, whether formally constituted or not. Again, work-groups are described in terms of their role within the organization and their characteristics.
4. Identify and describe task–object pairs. These are the tasks that must be performed, coupled with the objects that are used to perform them or to which they are applied.
5. Identify stakeholder needs. Stages 2–4 are described in terms of both the current system and the proposed system. Stakeholder needs are identified by considering the differences between the two. For example, if a stakeholder is identified as currently lacking a particular skill that is required in the proposed system then a need for training is identified.
6. Consolidate and check stakeholder requirements. Here the stakeholder needs list is checked against the criteria determined at earlier stages.

Open System Task Analysis (OSTA)

1. The primary task which the technology must support is identified in terms of users' goals.
2. Task inputs to the system are identified. These may have different sources and forms that may constrain the design.
3. The external environment into which the system will be introduced is described, including physical, economic and political aspects.
4. The transformation processes within the system are described in terms of actions performed on or with objects.
5. The social system is analyzed, considering existing work-groups and relationships within and external to the organization.

6. The technical system is described in terms of its configuration and integration with other systems.
7. Performance satisfaction criteria are established, indicating the social and tech- nical requirements of the system.
8. The new technical system is specified.

OSTA uses notations familiar to designers, such as data flow diagrams and textual descriptions.

Soft systems methodology

Clients – those who receive output or benefit from the system.

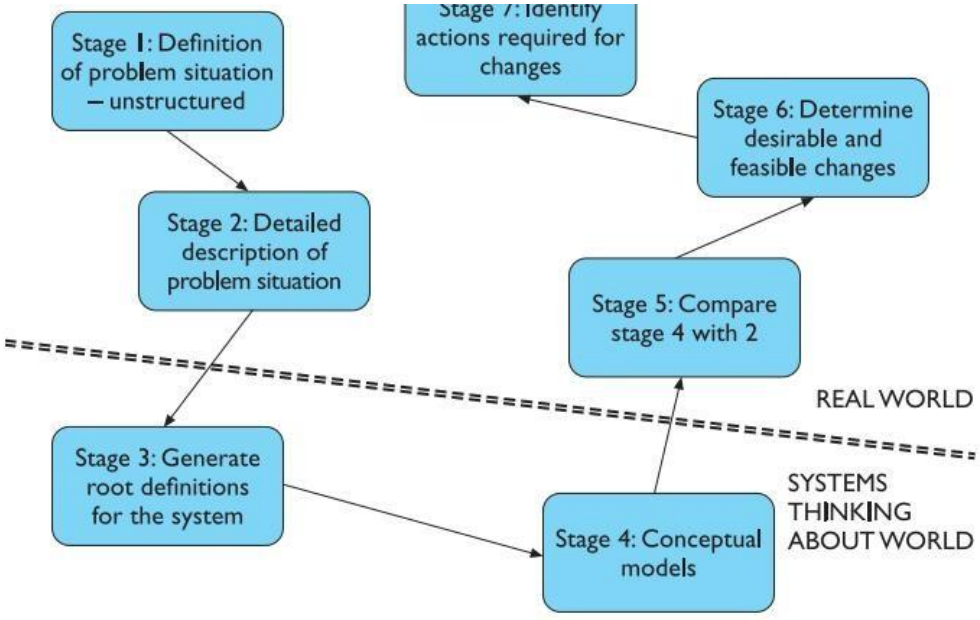
Actors – those who perform activities within the system.

Transformations – the changes that are effected by the system. This is a critical part of the root definition as it leads to the activities that need to be included in the next stage. These transform' the inputs of the system into the required outputs.

Weltanschauung – (from the German) meaning world view. This is how the system is perceived in a particular root definition.

Owner – those to whom the system belongs, to whom it is answerable and who can authorize changes to it.

Environment – the world in which the system operates and by which it is influenced



.

4. Explain how face to face communication is done.

FACE-TO-FACE COMMUNICATION

Face-to-face contact is the most primitive form of communication – primitive, that is, in terms of technology. If, on the other hand, we consider the style of communication, the interplay between

different channels and productivity, we instead find that face-to-face is the most sophisticated communication mechanism available.

Transfer effects and personal space

When we come to use computer-mediated forms of communication, we carry forward all our expectations and social norms from face-to-face communication. People are very adaptable and can learn new norms to go with new media (for example, the use of 'over' for turn-taking when using a walkie-talkie). However, success with new media is often dependent on whether the participants can use their existing norms.

An example of these problems concerns personal space. When we converse with one another we tend to stand with our heads a fairly constant distance apart. If people start to converse at opposite ends of a room, they will quickly move toward one another until they are a few feet apart. The exact distance depends somewhat on context; a high level of noise may make people come closer just to be heard. However, even in crowded rooms, conversants will dip their heads toward one A similar problem can occur in a video conference. Imagine Eustace and Bud have monitors with cameras mounted above, so that their offices are connected. The zoom on each camera is adjustable and Bud's camera is set with a wide focus, whilst Eustace's is set with a high level of zoom.

Eye contact and gaze

People who look away when you look at them may seem shifty and appear to be hiding something. Furthermore, relative frequency of eye contact and who 'gives way' from direct eye contact is closely linked to authority and power.

Gestures and body language

In a similar but more direct way, we use our hands to indicate items of interest. This may be conscious and deliberate as we point to the item, or may be a slight wave of the hand or alignment of the body. Again, a video connection may not be sufficientto allow our colleagues to read our movements.

Back channels, confirmation and interruption

It is easy to think of conversation as a sequence of utterances: A says something, then B says something, then back to A. This process is called *turn-taking* and is one of the fundamental

structures of conversation. However, each utterance is itself the result of intricate negotiation and interaction. Consider the following transcript:

Alison: Do you fancy that film . . . *er* . . . 'The Green' *um* . . . it starts at eight.

Brian: Great!

Turn-taking

In other cases, the speaker may explicitly *offer* the floor to the other participant. This may be in the form of a direct question, 'what do you think?', or simply a very strong change of tone.

5. Explain how communication is done through conversation.

Basic conversational structure

Imagine we have a transcript of a conversation, recalling from Chapter 9 that the production of such a transcript is not a simple task. For example, a slightly different version of Alison and Brian's conversation may look like this:
Alison: Do you fancy that film?
Brian: The *uh* (*500 ms*) with the black cat – 'The Green whatsit'?
Alison: Yeah, go at *uh* ...(*looks at watch – 1.2 s*)...20 to?
Brian: Sure.

The requirement of adjacency can be broken if the pair is interposed with other pairs for clarification, etc.:
Brian: Do you want some gateau?
Alison: Is it very fattening?
Brian: Yes, very.
Alison: And lots of chocolate?
Brian: Masses.
Alison: I'll have a big slice then.

Despite these difficulties, we see that the search for adjacency pairs forces us to examine closely the structure of the conversation. Whether such structures are really part of the conversation, or imposed by us upon it, is less clear. Later we shall see far more complex conversational structures.

Context

internal context – dependence on earlier utterances. For example, when Brian says 'masses' in the last transcript, this is meaningful in the light of Alison's question 'and lots of chocolate?'. This in turn is interpreted in the context of Brian's original offer of gateau. **external context** – dependence on the environment. For example, if Brian had said simply 'do you want one?', this could have meant a slice of gateau, or, if he had been holding a bottle, a glass of wine, or, if accompanied by a clenched fist, a punch on the nose.
Brian:(*Points*) That post is leaning a bit.
Alison: That's the one you put in.

Brian: The corner post is leaning a bit.

Alison: That's the one you put in.
Real speech, probably more than the written word, is full of

Topics, focus and forms of utterance

Given that conversation is so dependent on context, it is important that the participants have a shared focus. We have addressed this in terms of the external focus – the objects
that are visible to the participants – but it is also true of the internal focus of the
conversation.
Alison: Oh, look at your roses . . .
Brian: Mmm, but I've had trouble with greenfly.
Alison: They're the symbol of the English summer.
Brian: Greenfly?
Alison: No roses silly!

classified into three kinds [335]:
substantive directly relevant to the development of the topic;
annotative points of clarification, elaborations, etc.;
procedural talking about the process of collaboration itself.

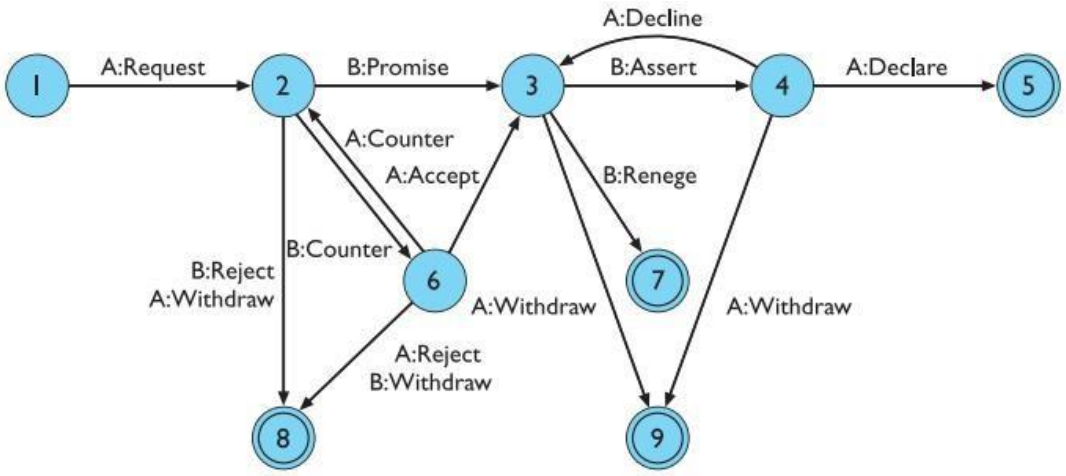
Breakdown and repair

At a lower level, we may see breakdown due to incorrectly read gestures or eyegaze, and through missed or inappropriate back channel responses. For instance, in Section 14.2.5, we described the problems in turn-taking during satellite-based video conferences. It may be difficult to interpret just where a breakdown occurred, as the breakdown may take some time to come to light, and be apparent at a different level from which it began. Alison and Brian are enjoying a day out at a country
park:

Constructing a shared understanding

A consequence of this model of conversation is that the participants are aware, at various levels both conscious and subconscious, that their common ground is incomplete. Their conversation is not then just an exchange of information about their task, but involves continual testing and cross-checking of the other party's understanding. Consider again a fragment from Alison's conversation about the way to the cinema:

Speech act theory



6.How the communication is dine using text based communication.

For *asynchronous* groupware (and even some synchronous systems), the major form of direct communication is text based. There are exceptions to this, for instance voice messaging systems and answerphones, and other media may be used in addition to text such as graphics, voice annotation or even video clips. But despite these, text is still the dominant medium. Text-based communication is familiar to most people, in that they will have written and received letters. However, the style of letter writing and that of face-toface communication are very different. The text-based communication in groupware systems

is acting as a speech substitute, and, thus, there are some problems adapting between the two media. There are four types of textual communication in current groupware:

discrete – directed message as in email. There is no explicit connection between different messages, except in so far as the text of the message refers to a previous one.

linear – participants' messages are added in (usually temporal) order to the end of a single transcript.

non-linear – when messages are linked to one another in a hypertext fashion. **spatial** – where messages are arranged on a two-dimensional surface.

Back channels and affective state

In addition to this loss of back channels, the speaker's tone of voice and body language are of course absent. These normally convey the *affective state* of the speaker (happy, sad, angry,

umorous) and the *illocutionary force* of the message (an important and urgent demand or a deferential request). Email users have developed explicit tokens of their affective state by the use of 'flaming' and 'smilies', using punctuation and acronyms; for example:

:-) – smiling face, happy

:-(– sad face, upset or angry

;-) – winking face, humorous

LOL – laughing out loud.

People tend to use stronger language in email than in face-to-face conversation,

Grounding constraints

cotemporality – an utterance is heard as soon as it is said (or typed); **simultaneity** – the participants can send and receive at the same time; **sequence** – the utterances are ordered. Linear transcripts obviously have some idea of sequence, but this is confused by the overlap and interleaving caused by the lack of cotemporality and simultaneity.

Consider this typical interchange during the use of the York Conferencer system:

1. Bethan: How many should be in the group?
2. Rowena: Maybe this could be one of the four strongest reasons?
3. Rowena: Please clarify what you mean.
4. Bethan: I agree.
5. Rowena: Hang on.
6. Rowena: Bethan what did you mean?

Context and deixis

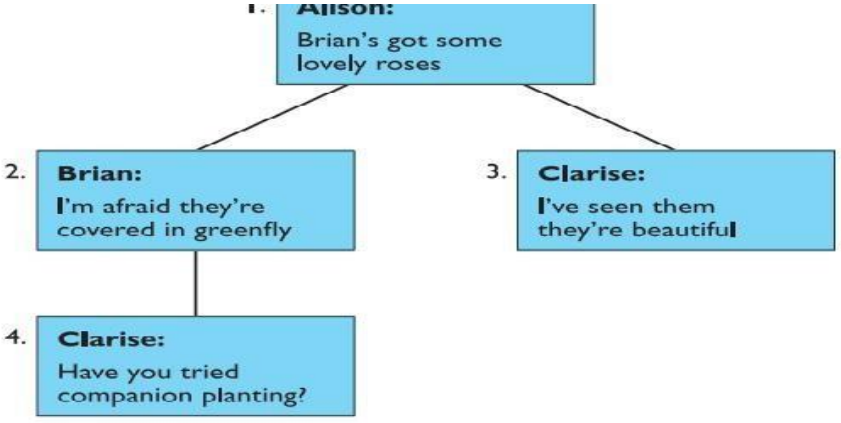


Figure 14.3 Hypertext conversation structure

7. How group work is done in communication and collaboration models.

Group dynamics

Even the naming of roles can cause problems. A person may be an author of a book or paper, but never write the words in it, acting instead as a source of ideas and comments. A particular case of this is the biographical story where the individual concerned and a professional writer co-author the book, but only the professional author writes. A co-authoring system such as Quilt would call the non-writing author a 'commentator' or a 'reviewer', but *not* an 'author'. One can imagine some of the social friction such naming will cause.

The group may also divide into subgroups for detailed discussion and then reform. Tools must be able to support this. For example, early versions of *CoLab's* software only catered for a single WYSIWIS screen – that is, they only supported a single group. Later versions were forced to allow subgroups to work independently

Physical layout3

The designers of Capture Lab, an eight-person meeting room, considered all these features and many other subtle effects. However, the users still had some difficulty in adapting to the *power positions* in the electronic meeting room. At first sight, the electronic meeting room is not unlike a normal conference room. If the shared screen is a whiteboard or an overhead projector, then the most powerful position is toward the front of the room (seats 1 or 6 in Figure 14.7). Managers would normally take this seat as they can then easily move to the whiteboard or overhead projector to point out some item and draw the group's attention.

Distributed cognition

Distributed cognition has profound effects on the way we look at group working and even individual work. It emphasizes the importance of *mediating representations*, for example the drawings on a whiteboard. These are no longer just a means of communicating between the parties, but can be a concrete embodiment of group knowledge. Furthermore, it constantly reminds us that communication is not just about getting knowledge from one person's head to another, but about the creation of new group knowledge, not necessarily grasped in totality by any single member.

8.Explain hypertext in detail.

Hypertext definition – text, hypertext

Hypertext attempts to get around these limitations of text by structuring it into a mesh rather than a line. This allows a number of different pages to be accessed from the current one, and, if the hypertext is well designed, the user should find it easier to follow his own particular idea through the mesh rather than being forced down one route. Typically, hypertext systems incorporate diagrams, photographs and other media as well as text. Such systems are often known as *multimedia* or *hypermedia* systems, although the three terms are often used interchangeably.
A hypertext system comprises a number of pages and a set of connect pages together. The links can join any page to any other page, and there can be more than one link per page. Thus a hypertext document does not simply start a linear progression and follow it to an end, but goes in lots of different directions, some of which terminate, while others link back into different parts of the document (see Figure 21.1, which illustrates the difference between linear text and hypertext).

There are many different ways of traversing the network, and so there are many different ways of reading a hypertext document .

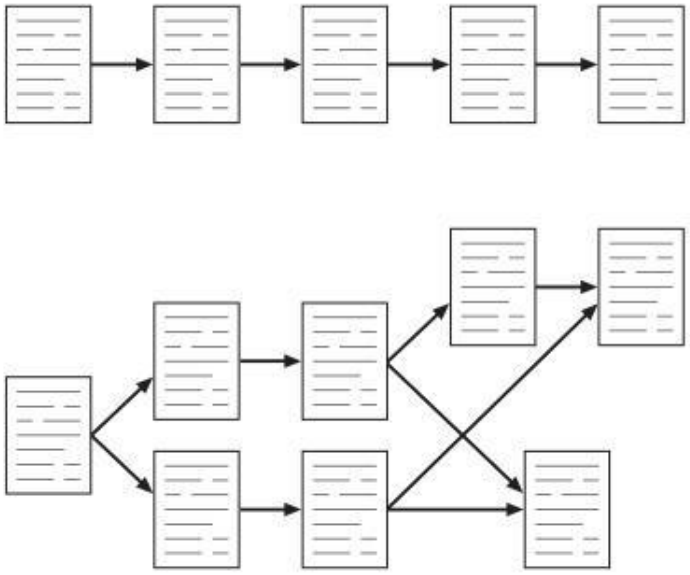


Figure 21.1 Typical structures of linear text and hypertext

21.2.2 Rich content

As well as static material – text and static diagrams and photographs – hypertext systems may also include more dynamic material such as animation, video and audio clips, and even full computer applications.

Animation

Animation is the term given to the addition of motion to images, making them move, alter and change in time. A simple example of animation in an interface is in the form of a clock. Digital clocks can flick by the seconds, whilst others imitate Salvador Dali and bend and warp one numeral into the next. Analog clocks have moving hour and minute hands, with an optional second hand sweeping round the clock face. Such a desktop accessory is found in a lot of interface setups, and the additional processing time required to produce such effects is no longer a major factor. Another common use for animation in current windowing systems is to animate

9.Explain in detail about multimedia.

Video and audio

In a media dominated world, there are strong arguments for using video or audio material as part of hypertext systems whether for education, entertainment or reference. Both audio and video material are expensive and time consuming to produce, but increasingly even home-PC systems include video and audio editing as standard. For example, the iMac includes a suite for editing video and burning DVDs.

Computation, intelligence and interaction

Of course the good thing about a computer is that not only can it *show* things that have previously been prepared, it can also *do* things. This book has an index, but it does not contain every word in the text (it may also refer to parts of the text that do not have a particular word). However, the web search can look through all the
chapters and find any words you want.

Delivery technology

On the computer

Some hypertexts, in particular help systems, are downloaded or installed permanently on a computer. This has the advantage of instant access and such applications need not use a standard viewer but may include their own bespoke browsing software. However, with media-rich hypertexts containing substantial graphics, video and audio clips it may be impractical to store everything on hard disk. Also, for copyright protection, some systems will deliberately not allow themselves to be copied from their original distribution media.on DVD and know that it could be used on any machine.

On the web

Of course, the *world wide web* is the best-known multimedia hypertext system of all. The world wide web offers a rich environment for the presentation of information. Documents can be constructed that are very different from paper versions; basic text can be augmented through the use of hypertext links to other documents, while graphics can easily be incorporated as pictures, photographs, icons, page dividing bars, or backgrounds. Pages can also have hypertext links embedded into different regions, which take the user to a different page or graphic if they are clicked on;

Application areas

Rapid prototyping

HyperCard on Macintosh computers has been very influential as a basis for experimental hypertext systems. HyperCard uses the metaphor of a card index, around which the user can navigate. Each card can hold text, diagrams, photographs,

Help and documentation

Hypermedia systems are ideally suited to online manuals and other help system applications (see Chapter 10). They allow user-oriented access to the information, and support browsing. In addition the information can be organized hierarchically, with successive selections providing more detailed information. This supports the varying needs that users have, such as quick reference, usage information, full details and so on. Many commercial help systems use hypermedia-style help. Good examples are the Sun Guide system, HyperCard help and Microsoft Windows help (used by many Windows applications). Educational systems are another common application. Hypermedia provides an

Education and e-learning

Hypertext and hypermedia are used extensively in educational settings, as they allow varied subjects to be related to each other in numerous ways so that the learner can investigate the links between different areas. In contrast to *computer-aided learning* (*CAL*) packages, hypermedia allows a student-controlled learning process.

10. Explain in detail about world wide web.

Basics

The web consists of a set of protocols built on top of the internet that, in theory, allow multimedia documents to be created and read from any connected computer in the world. The web supports hypertext, graphics, sound and movies, and, to structure and describe the information, uses a language called HTML (hypertext markup language) or in some cases, XML (extensible markup language). HTML is a markup language that allows hypertext links, images, sounds and movies to be embedded into text, and it provides some facilities for describing how these components are laid out. HTML documents are interpreted by a viewer, known as a rowser; there are many browsers, and each can interpret HTML in subtly different ways, or support different levels of functionality, which means that a web page viewed through one browser can look very different from the same page viewed through another. The web requires no particular multimedia capabilities from the machines that run the browsers; for example, if sound is unavailable on a particular machine, then obviously no sound is heard but the browser still displays the text happily.

Web technology and issues

As well as static web content such as text and images, many pages are *dynamic*: for

example, they may be generated from data held in databases, respond to individual information entered into forms, or include dynamic elements such as Java applets.

21.4.2 Web servers and web clients

Whereas a conventional PC program runs and is displayed on one computer, the web is *distributed*. Different parts of it run on different computers, often in different countries of the world. They are linked, of course, by the internet, an enormous global computer network . The pages are stored on web servers that may be on a company's own premises or in special data centers. Because they are networked, the webmaster for a site can upload pages to the server from wherever she is. For example, the web pages for www.hcibook.com are stored in a data center several thousand miles from where any of the authors live!

If the page contains images the same process is repeated for each image, and if the page is a framed one for each frame within the page.

21.4.3 Network issues

The fact that the web is networked raises a series of issues that can impact on usability.

Network capacity is called *bandwidth*. This is a measure of the amount of information that can pass down the channel in a given time. For example, a typical modem speed is 56 kbs – that is 56 kilobits per second. This equates to about 6000 characters per second. This sounds fine until you realize that images may take many tens or hundreds of characters (bytes) to encode...this is why many have renamed the web the 'world wide wait'! However, bandwidth is not the only important measure. There is also the time it takes for a message to get across the network from your machine to the web server nd back. This delay is called *latency*. Latency is caused by several factors – the finite

STATIC WEB CONTENT

The message and the medium

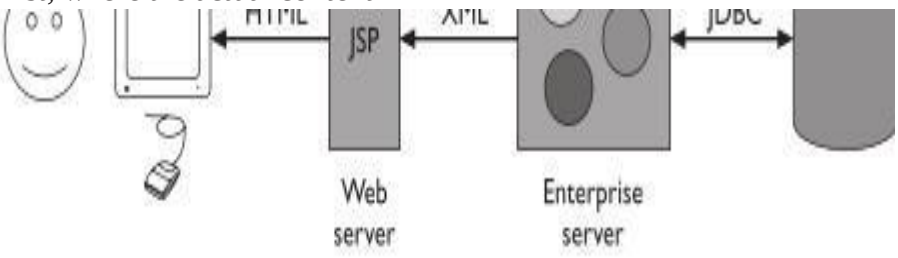
On the other hand, poor design can mean that excellent material is never seen by potential readers, as they have become bored, or intolerant of the medium, or confused, or for a host of other reasons have aborted their attempts to download and view the information. Pages do have to look immediately interesting and attractive if people are to spend time, effort and, because of the communication costs, money, in viewing them; the user-centered nature of the medium makes this imperative. This is in marked contrast to television or cinema or other

DYNAMIC WEB CONTENT

The active web

In the early days, the web was simply a collection of (largely text) pages linked together. The material was static or slowly changing and much of it authored and updated by hand. Some pages were generated on the fly, in particular the gateways into ftp servers and to gophers, which were so important in adding 'free' content to the web [97]. However, even here the user's model was still of a static repository of information. Web surfers may not have always known where they were, but they had a pretty good idea of what they were seeing and that if they came back it would be the same.

As HCI researchers and designers, we can neither ignore nor uncritically accept new technology in the web. The active web is here, our job is to understand it and to learn how to use it appropriately. In previous sections, we have already looked at the simplest form of active web page, those with movies, animated gifs or streaming audio. These are simplest, not in the sense that no effort is required – a short video clip may require many days of production effort – but in the sense that they have least user interaction. In this section we'll look at more complex forms of interaction. First, where the actual content



UNIT – IV
PART - A

1. Identify the categories of mobile platforms.(APR/MAY 2017)

A mobile platform's primary duty is to provide access to the devices. To run software and services on each of these devices, you need a platform, or a core programming language in which all of your software is written. Like all software platforms, these are split into three categories:

Licensed
Proprietary
open source.

2. Give examples of mobile design tools. .(APR/MAY 2017)

Mobile framework	**Design tool**
Android	Photoshop, XML-based themes
Palm webOS	Photoshop, HTML, CSS, and JavaScript
Mobile web	Photoshop, HTML, CSS, and JavaScript
Mobile widgets	Photoshop, HTML, CSS, and JavaScript
Mobile web apps	Photoshop, HTML, CSS, and JavaScript

3. List some of the open source platforms used for mobile HCI development.

Symbian OS
Android
Windows Mobile
Palm OS
Linux
Mac OS X

4. What is web runtimes?

- Installs a launch icon in the native operating system.
- Can verify and then launch an app from the native operating system in a frameless window.
- Shows the app in the native environment as a typical running application (so that application switching and monitoring work as expected).
- Integrates with device capabilities.
- The WebAPI project is progressing towards supporting these APIs directly in Gecko
- Provides a seamless launch experience. When an app is launched, it can check that its receipt is valid (if the app uses a receipt).

5. What Is Mobile 2.0?

Mobile 2.0, refers to a perceived next generation of mobile internet services that leverage the social web, or what some call Web 2.0. The social web includes social networking sites and wikis that emphasise collaboration and sharing amongst users. Mobile Web 2.0, with an emphasis on Web, refers to bringing Web 2.0 services to the mobile internet, Mobile 2.0 refers to services that integrate the social web with the core aspects of mobility personal, localized, always-on and ever-present.

6. Define Typography.

The term typography is also applied to the style, arrangement, and appearance of the letters, numbers, and symbols created by the process. Type design is a closely related craft, sometimes considered part of typography; most typographers do not design typefaces, and some type designers do not consider themselves typographers.

7. List pros and cons of SMS. The pros of SMS applications

- They work on any mobile device nearly instantaneously.
- They're useful for sending timely alerts to the user.
- They can be incorporated into any web or mobile application.
- They can be simple to set up and manage.

The cons of SMS applications include:

- They're limited to 160 characters.

They provide a limited text-based experience.

- They can be very expensive.

8. What are mobile web widgets?

A web widget is a software widget for the web. It's a small application with limited functionality that can be installed and executed within a web page by an end user. A widget has the role of a transient or auxiliary application, meaning that it just occupies a portion of a webpage and does something useful with information fetched from other websites and displayed in place. Other terms used to describe web widgets include: portlet, web part, gadget, badge, module, snippet and flake. Widgets are typically created in DHTML or Adobe Flash.

9. List some game applications in mobile HCI.

Mobile games tend to be small in scope and many priorities innovative design and ease of play over visual spectacle. Storage and memory limitations place constraints on file size that presently rule out the direct migration of many modern PC and console games to mobile.

Location-based mobile games

Multiplayer mobile games

10. What is meant by mobile information architecture?

The mobile information architecture defines not just how your information will be structured, but also how people will interact with it. This is made especially tricky when you consider that different devices have different capabilities and therefore different interaction models. Take the way people interact with their devices: for example, a touch device on which the user literally points and clicks, or a more basic device on which the user uses the directional pad to navigate to the desired location.

11. What is meant by wireframes

Wireframes are a way to lay out information on the page, also referred to as information design. Site maps show how our content is organized in our informational space; wireframes show how the user will directly interact with it. Wireframes are like the peanut butter to the site map jelly in our information architecture sandwich.

PART - B

1. Appraise the type of mobile applications with examples. (APR/MAY 2017)

Mobile Application Medium Types

SMS

The most basic mobile application you can create is an SMS application. Although it might seem odd to consider text messages applications, they are nonetheless a designed experience. Given the ubiquity of devices that support SMS, these applications can be useful tools when integrated with other mobile application types. Typically, the user sends a single keyword to a five-digit short code in order to return information or a link to premium content. For example, sending the keyword "freebie" to a hypothetical short code "12345" might return a text message with a coupon code that could be redeemed at a retail location, or it could include a link to a free ringtone.

Pros

The pros of SMS applications include:

• They work on any mobile device nearly instantaneously.

They're useful for sending timely alerts to the user.

• They can be incorporated into any web or mobile application.

• They can be simple to set up and manage.

Cons

The cons of SMS applications include:

• They're limited to 160 characters.

• They provide a limited text-based experience.

• They can be very expensive.

Mobile Websites

Mobile websites often have a simple design and are typically informational in nature, offering few—if any—of the interactive elements you might expect from a desktop site. Mobile websites have made up the majority of what we consider the mobile web for the past decade, starting with the early WML-based sites (not much more than a list of

links) and moving to today's websites, with a richer experience that more closely resembles the visual aesthetic users have come to expect with web content.

Mobile Web Widgets

Largely in response to the poor experience provided by the mobile web over the years, there has been a growing movement to establish mobile widget frameworks and platforms. For years the mobile web user experience was

Severely underutilized and failed to gain traction in the market, so several operators, device makers, and publishers

Began.

Mobile Web Applications

Mobile web applications are mobile applications that do not need to be installed or compiled on the target device. Using XHTML, CSS, and JavaScript, they are able to provide an application-like experience to the end user while running in any mobile web browser. By "application-like" experience, I mean that they do not use the drill-down or page metaphors in which a click equals a refresh of the content in view. Web applications allow users to interact with content in real time, where a click or touch performs an action within the current view.

Native Applications

The next mobile application medium is the oldest and the most common; it is referred to as native applications, which is actually a misnomer because a mobile web app or mobile web widget can target the native features of the device as well. These applications actually should be called "platform applications," as they have to be developed

And compiled for each mobile platform.

Games

These differences, in my mind, are what make mobile games stand apart from all other pplication genres—their capability to be unique and difficult to duplicate in another application type, though the game itself is relatively easy to port. Looking at this model for other application areas—namely, the mobile web—could provide helpful insight into how we create the future of mobile web applications.

2 .List and explain the elements of mobile design.(APR/MAY 2017)

Context

Who are the users? What do you know about them? What type of behavior can you assume or predict about the users?

• What is happening? What are the circumstances in which the users will best absorb the content you intend to present?

• When will they interact? Are they at home and have large amounts of time? Are they at work where they have short periods of time? Will they have idle periods of time while waiting for a train, for example?
• Where are the users? Are they in a public space or a private space? Are they inside or outside? Is it day or is it night?
• Why will they use your app? What value will they gain from your content or services in their present situation?
• How are they using their mobile device? Is it held in their hand or in their pocket?
How are they holding it? Open or closed? Portrait or landscape?

Message
Another design element is your message, or what you are trying to say about your site or application visually. One might also call it the "branding," although I see branding and messaging as two different things. Your message is the overall mental impression you create explicitly through visual design. I like to think of it as the holistic or at times
instinctual reaction someone will have to your design. If you take a step back, and look at a design from a distance, what is your impression? Or conversely, look at a design for 30 seconds, and then put it down. What words would you use to describe the experience?Branding shouldn't be confused with messaging. Branding is the impression your com

Look and Feel
The concept of "look and feel" is an odd one, being subjective and hard to define. Typically, look and feel is used to describe appearance, as in "I want a clean look and feel" or "I want a usable look and feel." The problem is: as a mobile designer, what does it mean?

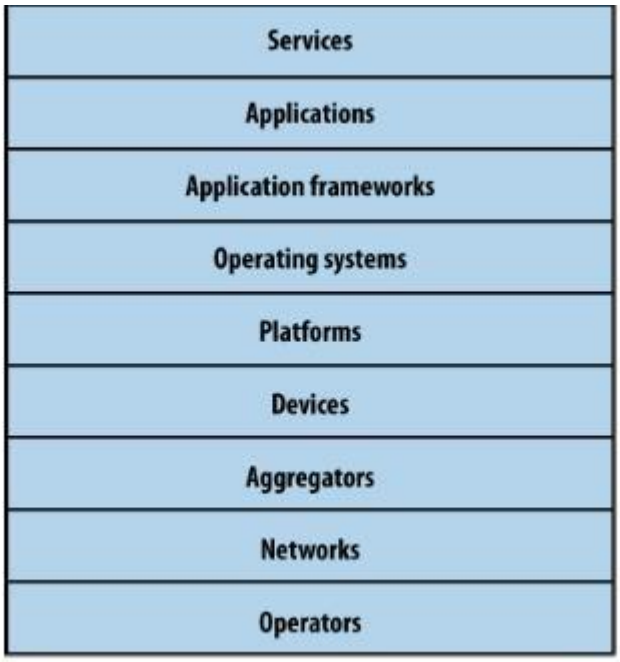
Layout
Layout is an important design element, because it is how the user will visually process the page, but the structural and visual components of layout often get merged together, creating confusion and making your design more difficult to produce.

Color
The fifth design element, color, is hard to talk about in a black-and-white book. Maybe it is fitting, because it wasn't that long ago that mobile screens were available only in black and white (well, technically, it was black on a green screen). These days, we have nearly the entire spectrum of colors to choose from for mobile designs.

Typography
The sixth element of mobile design is typography, which in the past would bring to mind the famous statement by Henry Ford: Any customer can have a car painted any color that he wants so long as it is black.

Graphics
The final design element is graphics, or the images that are used to establish or aid a visual experience. Graphics can be used to supplement the look and feel, or as content displayed inline with the text.

3. **Explain mobile ecosystem**.



4.Explain mobile information architecture.

What Is Information Architecture?

. This definition outlines the following:

The structural design of shared information environments

The combination of organizations, labeling, search, and navigation systems within websites and intranets

• The art and science of shaping information products and experiences to support usability and findability

• An emerging discipline and community of practice focused on bringing principles of design and architecture to the digital landscape Similar to how mobile technology has many facets, so does information architecture, as it is often used as an umbrella term to describe several unique disciplines, including the following:

Information architecture

The organization of data within an informational space. In other words, how the user will get to information or perform tasks within a website or application.

Interaction design

The design of how the user can participate with the information present, either in a direct or indirect way, meaning how the user will interact with the website of application to create a more meaningful experience and accomplish her goals.

Information design

The visual layout of information or how the user will assess meaning and direction given the information presented to him.

Navigation design

The words used to describe information spaces; the labels or triggers used to tell the users what something is and to establish the expectation of what they will find.

Interface design

1. The design of the visual paradigms used to create action or understanding.

Clear, simple labels

Good trigger labels, the words we use to describe each link or action, are crucial in Mobile. Words like "products" or "services" aren't good trigger labels. They don't tell us anything about that content or what we can expect. Now, I would argue that good trigger labels are crucial in the Web as well, that we've become lazy and we assume so much about the user that we ignore the use of good trigger labels. Users have a much higher threshold of pain when clicking about on a desktop site or application, hunting and pecking for tasty morsels. Mobile performs short, to-the-point, get-it-quick, and get-out types of tasks. What is convenient on the desktop might be a deal breaker on mobile.

2. **Site Maps**

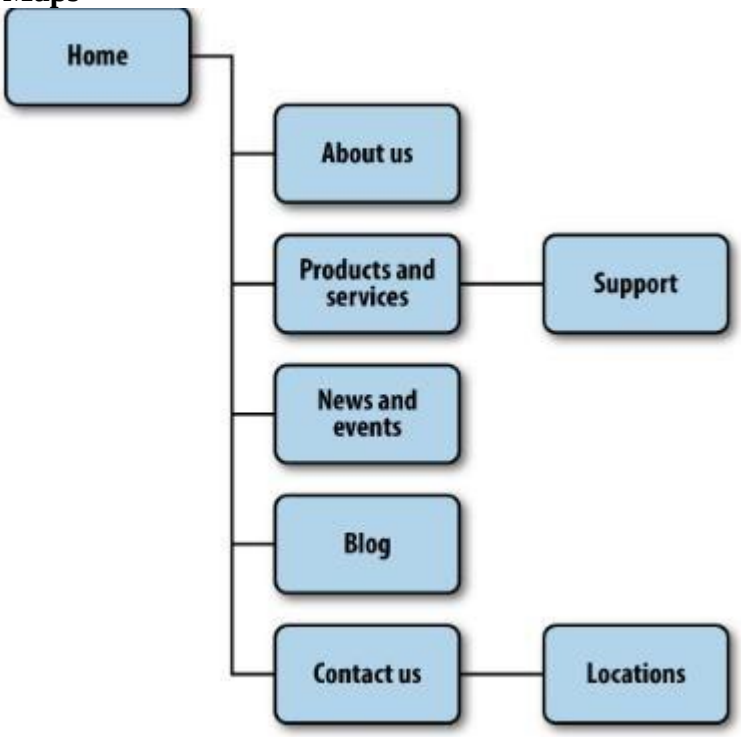


Figure 7-6. Teasing content to confirm the user's expectations of the content within

Clickstreams

Clickstream is a term used for showing the behavior on websites, displaying the order in which users travel through a site's information architecture, usually based on data gathered from server logs. Clickstreams are usually
h istorical, used to see the flaws in your information architecture, typically using heat-mapping or simple percentages to show where your users are going. I've always found them to be a useful tool for rearchitecting large websites.

Wireframes

The next information architecture tool at our disposal is wireframes. Wireframes are a way to lay out information on the page, also referred to as information design. Site maps show how our content is organized in our informational space; wireframes show how the user will directly interact with it.

5. **Whata are the types of prototyping used in mobile information architecture.**

Prototyping

Prototypes might sound like a scary (or costly) step in the process. Some view them as redundant or too time-consuming, preferring to jump in and start coding things. But as with wireframes, I've found that each product we've built out some sort of prototype has saved both time and money. The following sections discuss some ways to do some simple and fast mobile prototyping.

Paper prototypes

The most basic level we have is paper prototyping: taking our printed-out wireframes or even drawings of our interface, and putting them in front of people.

Context prototype

The next step is creating a context prototype .Take a higher-end device that enables you to load full-screen images on it. Take your wireframes or sketches and load them onto the device, sized to fill the device screen. Leave the office. Go for a walk down to your nearest café. Or get on a bus or a train. As you are traveling about, pull out your device and start looking your interface in the various contexts you find yourself currently in.

HTML prototypes

The third step is creating a lightweight, semifunctional static prototype using XHTML, CSS, and JavaScript, if available. This is a prototype that you can actually load onto a device and produce the nearest experience to the final product, but with static dummy content and data .It takes a little extra time, but it is worth the effort.

6.**Expalin the mobile design tools.**

Table 8-4. Design tools and interface toolkits

Mobile framework	Design tool	Interface toolkits
Java ME	Photoshop, NetBeans	JavaFX, Capuchin
BREW	Photoshop, Flash	BREW UI Toolkit, uiOne, Flash
Flash Lite	Flash	Flash Lite
iPhone	Photoshop, Interface Builder	iPhone SDK
Android	Photoshop, XML-based themes	Android SDK
Palm webOS	Photoshop, HTML, CSS, and JavaScript	Mojo SDK
Mobile web	Photoshop, HTML, CSS, and JavaScript	W3C Mobile Web Best Practices
Mobile widgets	Photoshop, HTML, CSS, and JavaScript	Opera Widget SDK, Nokia Web Runtime
Mobile web apps	Photoshop, HTML, CSS, and JavaScript	iUI, jQTouch, W3C Mobile Web App Best Practices

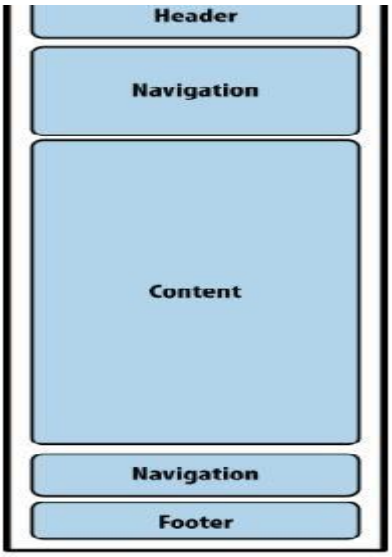
Designing for the Right Device

With the best possible experience at hand, take a moment to relish it. Remind yourself that you are working with a rapidly evolving medium and though it might not be possible for every user to experience things exactly the way you've intended, you've set the tone and the vision for how the application should look. The truly skilled designer

doesn't create just one product—she translates ideas into experiences. The spirit of your design should be able to be adapted to multiple devices.

Designing for Different Screen Sizes

Mobile devices come in all shapes and sizes. Choice is great for consumers, but bad for design. It can be incredibly difficult to create that best possible experience for a plethora of different screen sizes. For example, your typical feature phone might only be 140 pixels wide, whereas your higher-end smartphone might be three to four times wider.



7.What is mobile 2.0 .Explain in detail.

The mobile community started to discuss the idea of "Mobile 2.0," borrowing from many of the same principles behind Web 2.0. Each of these principles serves to transform the Web into a more agile and user-centered medium for delivering information to the masses. Mobile development, under the bottlenecks of device fragmentation and operator control, is sorely in need of a little reinvention as well.

Following is a recap of the original seven principles of Web 2.0:

The Web as a platform

For the mobile context, this means "write once, deploy everywhere," moving away from the costly native applications deployed over multiple frameworks and networks.

Harnessing collective intelligence

This isn't something the mobile community has done much of, but projects like WURFL—an open source repository of device profiles provided by the community—is exactly what mobile needs more of.

Data is the next Intel inside

Mobile takes this principle several steps further. It can include the data we seek,

the data we create, and the data about or around our physical locations.

End of the software release cycle

Long development and testing cycles heavily weigh on mobile projects, decreasing all hopes of profitability. Shorter agile cycles are needed to make mobile development work as a business. Releasing for one device, iterating, improving, and then releasing for another is a great way to ensure profitability in mobile.

Lightweight programming models

Because mobile technology is practically built on enterprise Java, the notion of using lightweight models is often viewed with some skepticism. But decreasing the programming overhead required means more innovation occurs faster. *Software above the level of a single device* This effectively means that software isn't just about computers anymore. We need to approach new software as though the user will demand it work in multiple contexts, from mobile phones to portable gaming consoles and e-book readers.

Rich user experiences

A great and rich user experience helps people spend less time with the software and more time living their lives. Mobile design is about enabling users to live their lives better.

8.Explain in detail about mobile application framework with example.

Java

Applications written in the Java ME framework can often be deployed across the majority of Java-based devices, but Given the diversity of device screen size and processor power, cross-device deployment can be a challenge. Most Java applications are purchased and distributed through the operator, but they can also be downloaded and installed via cable or over the air.

S60

The S60 platform, formerly known as Series 60, is the application platform for devices that run the Symbian OS. S60 is often associated with Nokia devices—Nokia owns the platform—but it also runs on several non-Nokia devices. S60 is an open source framework.

BREW

Applications written in the BREW application framework can be deployed across the majority of BREW-based devices, with slightly less cross-device adaption than other frameworks.

Flash Lite

Adobe Flash Lite is an application framework that uses the Flash Lite and ActionScript frameworks to create vector-based applications. Flash Lite applications can be run within the Flash Lite Player, which is available in a handful of devices around the world.

Windows Mobile

Applications written using the Win32 API can be deployed across the majority of Windows Mobile-based devices. Like Java, Windows Mobile applications can be downloaded and installed over the air or loaded via a cable-connected computer.

Cocoa Touch

Cocoa Touch is the API used to create native applications for the iPhone and iPod touch. Cocoa Touch applications must be submitted and certified by Apple before being included in the App Store. Once in the App Store, applications can be purchased, downloaded, and installed over the air or via a cable-connected computer.

Android SDK

The Android SDK allows developers to create native applications for any device that runs the Android platform. By using the Android SDK, developers can write applications in C/C++ or use a Java virtual machine included in the OS that allows the creation of applications with Java, which is more common in the mobile ecosystem.

WebKit

With Palm's introduction of webOS, a mobile platform based on WebKit, and given its predominance as a mobile browser included in mobile platforms like the iPhone, Android, and S60, and that the vast majority of mobile web apps are written specifically for WebKit, I believe we can now refer to WebKit as a mobile framework in its own
right.

9.**Explain the types of mobile context with example.**

Application Context

Applications can be presented in a variety of ways, ranging from a simple task-based utility to an experience meant to consume the user's focus and attention. There of course is no right or wrong direction—only what is best for your user. In fact, nothing says that you can't use multiple application contexts within the same application—I just wouldn't recommend it unless you have really thought out the flow of your application, because typically It is best to present only one application context so as to avoid confusing the user. If you think it best for your app to mix contexts, then give the user the option to switch between them; for example, some smartphones allow for an Orientation change, so if the device is rotated to landscape mode, your app switches from an informative view to a utility view, or maybe from a locale view to an immersive view.

Utility Context

The most basic application context is the utility, or a simple user experience metaphor that is meant to address short, task-based scenarios. Information is meant to be pre- sented in a minimal fashion, often using the least amount of user input as possible. An example of a utility might be a calculator, weather forecast, unit conversion, stocks, world clock, and so on. In each of these cases, the user enters a small amount of information into the utility, like a simple equation, a city, or a stock symbol, and either by performing a small background task or fetching information online, the utility is able to present data to the user in the desired context .

Locale Context

The locale context is a newer application type that is still being defined by the mobile community, but we are certainly seeing some clear patterns of how to create locale applications . As more location information is being published online, and more devices add GPS to pinpoint the user's location, locale is becoming an excellent
data point to pivot information around. For example, I can use location to display the cafés nearest to my current location. Plus I can layer multiple data sources into the application, such as: of the cafés nearest to me, which ones have free wireless access? Or, do I have friends in the general area that can meet me?

Productivity Application Context

They are able to make the same time commitment as they would in the desktop context. Productivity applications are often very structured, presenting information in a defined hierarchy and often using the folder or group metaphor to define a sense of order to the user. When designing these types of apps, we need to pay careful consideration to how the user thinks out the task. People have an uncanny ability to understand and recall complex hierarchies of tasks—for example, what they need to do first, second, and third in order for a particular solution to work. We take this for granted and in the desktop context often show the users the entire hierarchy visually. In the mobile context, we don't have the screen real estate, and therefore need to help users find their way.

10.What are the elements of mobile design.

The first is a natural gift for being able to see visually how something should look that produces a desired emotion with the target audience. The second is the ability to manifest that vision into something for others to see, use, or participate in. The third is knowing how to utilize the medium to achieve your design goals.

Context

Who are the users?
What do you know about them?
What type of behavior can you assume or predict about the users?
What is happening?
What are the circumstances in which the users will best absorb
the content you intend to present?
When will they interact?
Are they at home and have large amounts of time?
Are they at work where they have short periods of time?
Will they have idle periods of time while waiting for a train, for example?
Where are the users?
Are they in a public space or a private space?
Are they inside or outside?
Is it day or is it night?
Why will they use your app?
What value will they gain from your content or services in their present situation?
How are they using their mobile device?
Is it held in their hand or in their pocket?
How are they holding it? Open or closed?
Portrait or landscape?

Message

I'm not exactly sure what it is saying. Words you might use to describe the message are crisp, clean, and sharp.

ESPN The ESPN site clearly is missing a message. It is heavily text-based, trying to put a lot of content above the fold, but doesn't exactly deliver a message of any kind. If you took out the ESPN logo, you likely would have indifferent expectations of this site; it could be about anything, as the design doesn't help set expectations for the user in any way. Words you might use to describe the message: bold, cluttered, and content-heavy.

Disney

Disney creates a message with its design. It gives you a lot to look at—probably too much—but it clearly tries to say that the company is about characters for a younger audience. Words you might use to describe the message: bold, busy, and disorienting.

Wikipedia

The Wikipedia design clearly establishes a message. With a prominent search and text-heavy layout featuring an article, you know what you are getting with this design. Words you might use to describe the message: clean, minimal, and text-heavy.

Amazon

Amazon sort of creates a message. Although there are some wasted opportunities above the fold with the odd ad placement, you can see that it is mostly about products (which is improved even more if you scroll down). Words you might use to describe the message: minimal but messy, product-heavy, and disorienting.

Look and Feel

Layout

Layout is an important design element, because it is how the user will visually process the page, but the structural and visual components of layout often get merged together, creating confusion and making your design more difficult to produce. The first time layout should rear its head is during information architecture. In fact, I prefer to make about 90 percent of my layout decisions during the information architecture period. I ask myself questions like: where should the navigation go on the page or screen? What kind of navigation type should I use?Should I use tabs or a list? What about a sidebar for larger screens? All of these should be answered when defining the information architecture and before you begin to design.

Fixed versus fluid

Another layout consideration is how your design will scale as the device orientation changes, for example if the device is rotated from portrait mode to landscape and vice versa. This is typically described as either being fixed (a set number of pixels wide), or fluid (having the ability to scale to the full width of the screen regardless of the device orientation).

UNIT – V

PART – A

1. **What are the types of overlays? APR/MAY 2017**

   Input overlay
   Dialog overlay
   Detail overlay

2. **What is auto complete pattern?APR/ MAY 2017**

   Auto complete is a powerful pattern that benefits from a reactive interface. As the user types input into a field, a drop down menu of matching value is displayed. When done right, the choice that best matches will be auto selected. The user can stop typing answer accept the choice that has been matched or choose a different value from the list. The selected value is then entered into the field.

3. **What are the Events resent in drag and drop?**

   Page Load
   Mouse Hover
   Mouse Down
   Drag Initiated
   Drag Leaves Original Location
   Drag Re-Enters Original Location
   Drag Enters Valid Target
   Dragging over a valid drop target.

4. **Who are the Actors in drag and drop?**

   During each event you can visually manipulate a number of actors. The page elements available include:

- Page (e.g., static messaging on the page)
- Cursor
- Tool Tip
- Drag Object (or some portion of the drag object, e.g., title area of a module)
- Drag Object‟s Parent Container
- Drop Target

5. **What is meant by Toggle Selection?**

   Toggle Selection: Checkbox or control-based selection.

Checkboxes and toggle buttons are the familiar interface for selecting elements on most web pages. The way to select an individual mail message is through the row‟s checkbox. Clicking on the row itself does not select the message.

6. **What is meant by Scrolling versus paging?**

   A scrolling page layout is not better than paging when it comes to survey participation and survey completion rates. Conversely, paging is not better than scrolling when it comes to item non-response.
   Scrolling does allow people to complete surveys faster and with fewer technical difficulties; therefore, respondents evaluate their experiences with scrolled surveys more highly.
   Both text messages and e-mail are effective in recruiting people to complete online surveys, but those responding to text-messages have higher break-off rates.

There are no differences when it comes to laying out answer scales vertically versus horizontally, so either method will work.

7. What is meant by Interaction in Context?

The topics in this section provide an overview of support for Interaction Context in application development, Interaction Context enables the applications and UI frameworks to support multiple, concurrent interactions by providing gesture detection and manipulation processing.

8. Define Fitts's Law

Fitts's law is a predictive model of human movement primarily used in human computer interaction and ergonomics. This scientific law predicts that the time required to rapidly move to a target area is a function of the ratio between the distance to the target and the width of the target. Fitts's law is used to model the act of pointing, either by physically touching an object with a hand or finger, or virtually, by pointing to an object on a computer monitor using a pointing device.

9. What is meant by Dialog Overlay?

Dialog Overlay contains important information that the user should not ignore. Both the Netflix Purchase dialog and the Flicker Rotate dialog are good candidates for the Lightbox Effect

10. What is meant by Virtual Scrolling?

The virtual scrolling feature of datagrid can be used to display large amounts of records without paging. When scrolling with the vertical scrollbar, the datagrid executes ajax requests to load and refresh the existing records. The overall behavior is smooth and with no flicker.

11. What is meant by Zoomable User Interface?

A Zoom able User Interface (ZUI) way to create a virtual canvas. ZUI interactions is in a concept demo for Firefox on the mobile ZUIs provide the ultimate virtual canvas. By extending the concept of the page, the user never actually leaves the virtual page.

12. What is meant by Google Blogger?

Blogger is a blog-publishing service that allows multi-user blogs with time-stamped entries. It was developed by Pyra Labs, which was bought by Google in 2003. Generally, the blogs are hosted by Google at a subdomain of blogspot.com. Blogger is Google‟s free tool for creating blogs. It can be found on the web at http://www.blogger.com.

PART - B

1. Summarize the principles for designing rich web interfaces.(APR/MAY 2017)

The Principles

The six principles are simple and straightforward.

Make It Direct

The principle of WYSIWIG (What You See Is What You Get) has been proven over and over again during the last quarter of a century. Allowing users to directly edit content in context (In-Page Editing), control the interface with Drag and Drop, and directly manipulate objects (Direct Selection) all work toward creating an intelligent interface.

Keep It Lightweight

Respecting the user's level of effort is key to producing an effortless interface. Understanding the user's intent and providing just the right amount of interface (Contextual Tools) within the current context is critical to providing a lightweight experience.

Stay on the Page

Instead of breaking into the user's normal workflow with repeated page refreshes, we can now create an experience that more closely matches the user's flow. The proper use of Overlays, Inlays, Virtual Space, and Process Flows is integral to matching the way the user wants to work—not the way we forced them to work in the past.

Provide an Invitation

With an array of interactions at our disposal, it would be easy to have most of our features go unnoticed and unused. Throwing out contextual "welcome mats" within the page both statically and dynamically (Affordance Invitation, Call to Action Invitation, Blank Slate Invitation, Tour Invitation, Hover Invitation, Drag and Drop Invitation, Inference Invitation, and More Content Invitation) invites users to explore new idioms, improving their overall experience.

Use Transitions

Not just for those annoying mortgage ads, Transitions are necessary for both communication and engagement. With a wide variety of cinematic effects at our disposal (Brighten and Dim, Expand/Collapse, Self-Healing, Animation, Spotlight, Lightbox Effect, Faceplate, Flip, Carousel, Accordion, Slide In and Slide Out, and Zoom), we can either overwhelm our user with Animation Gone Wild or Needless Fanfare, or we can use these effects to explain happenings, show relationships, focus attention, improve performance, and create the illusion of virtual space.

Be Reactive

For every action there should be an equal and opposite reaction. This is the physics of our interfaces. Providing Auto Complete, Live Suggest, Live Search, Refining Search, Live Previews, Progressive Disclosure, Progress Indicators, and Periodic Refresh form the tools for creating a lively, reactive interface.

Staying Up to Date

The Web is constantly evolving, and it is impossible to always have the most up-to-date examples, and to capture emerging patterns in a book. We provide two resources to keep this work updated and relevant: *Designing Web Interfaces companion site* Please visit *http://designingwebinterfaces.com* for up-to-date information on this
book as well as updated examples, principles, and patterns. You can also contact the authors at this site.

2.Design a web interface for a ''library management system".State the functional requirements you are considering. .(APR/MAY 2017)

What are data gathering techniques?

Data gathering is a very important activity when requirement specification and correspondingevaluation is required. It is a process of preparing and collecting data. The purpose of datagathering is to obtain information to keep on record, to make decisions about important issues, pass information on to others. Data is gathered to provide information regarding a specific topic.Mainly there are five techniques for collecting or gathering data:

Interview

Questionnaires
Workshop or Focus group
Naturalistic Observation
Studying Documentation

Data gathering techniques selected for Library management system

Interview

The interview plays a vital role for gathering information from user because it involves face to facecontact with the user. In this technique, questions are put in front of the users and it require immediate response to view user requirement.Interview:

Can be used for evaluating information needs

Can be used for evaluating staff responsiveness, attitudes and perceptions of the library andinfo services.

Can be used for tracking the flow of information within the company students: Student has the lowest priority as compare to librarian student can use the systemin order to view the detail of books available.

Novice Users
Professionals

Secondary Stakeholders:

These people usually concern themselves with the reports, etc. They areassociated to the system through some primary stakeholder i.e. indirectly associated. According toour system under this category the users are:

Librarian

Librarian has the highest priority in library management system it act as admin of thesystem. Register student as user of the system. These are the following step:

Add and delete student and book record.
Search and edit the record of student.
Issue and return books.
Calculate fine
Other book Manufacturing Companies
Brokers and Agents
Small book shop holders

Tertiary Stakeholders:

Those who do not fall into Primary or

Keeping the feedback of users in mind we conclude the following:

- The library management system according to the user's tasks and fulfill all the requirements which are need.
- Provide sufficient feedback from user.
- The library management system would be easy to use and easy to learn.
- Proper metaphor is used to make it more simple and easy to use the system for different type of user who wants to use the system.

3.Explain the events handled in drag and drop .

There are at least 15 events available for cueing the user during a drag and drop interaction:

Page Load

Before any interaction occurs, you can pre-signify the availability of drag and drop.

For example, you could display a tip on the page to indicate draggability.

Mouse Hover

The mouse pointer hovers over an object that is draggable.

Mouse Down

The user holds down the mouse button on the draggable object.

Drag Initiated

After the mouse drag starts (usually some threshold—3 pixels).

Drag Leaves Original Location

After the drag object is pulled from its location or object that contains it.

Drag Re-Enters Original Location

When the object re-enters the original location.

Drag Enters Valid Target

Dragging over a valid drop target.

Drag Exits Valid Target

Dragging back out of a valid drop target.

Drag Enters Specific Invalid Target Dragging over an invalid drop target.

Drag Is Over No Specific Target

Dragging over neither a valid or invalid target. Do you treat all areas outside of valid targets as invalid?

Drag Hovers Over Valid Target

User pauses over the valid target without dropping the object. This is usually when a spring loaded drop target can open up. For example, drag over a folder and pause, the folder opens revealing a new area to drag into.

Drag Hovers Over Invalid Target

User pauses over an invalid target without dropping the object. Do you care? Will you want additional feedback as to why it is not a valid target?

Drop Accepted

Drop occurs over a valid target and drop has been accepted.

Drop Rejected

Drop occurs over an invalid target and drop has been rejected. Do you zoom back the dropped object? Drop on Parent Container

4.Explain how the direct selection method is involved in web interface design.

Toggle Selection
Checkbox or control-based selection.
Collected Selection
Selection that spans multiple pages.
Object Selection
Direct object selection.
Hybrid Selection
Combination of Toggle Selection and Object Selection

The above are the methods involved in direct selection and with example diagram it is implemented.

5. Explain the contextual tools in designing web interface

Most desktop applications separate functionality from data. Menu bars, toolbars, and palettes form islands of application functionality. Either the user chooses a tool to use on the data or makes a selection and then applies the tool.

Early websites were just the opposite. They were completely content-oriented. Rich tool sets were not needed for simply viewing and linking to content pages. Even in e-commerce sites like Amazon or eBay, the most functionality needed was the hyperlink and “Submit” button. However, this simplistic approach no longer exists in the current web application landscape. As the Web has matured, a wide variety of application styles has emerged. On one end of the spectrum there are simple sites that need no more functionality than the hyperlink and a “Submit” button. On the other end of the spectrum there are full applications hosted as a website.

Fitts's Law

Fitts's Law is an ergonomic principle that ties the size of a target and its contextual proximity to ease of use. Bruce Tognazzini restates it simply as:

The time to acquire a target is a function of the distance to and size of the target.

Contextual Tools

We could simply isolate our functionality into islands of tools (toolbars and menus). But this would work against Fitts's Law by requiring more effort from the user. It would also add more visual weight to the page. Instead of interacting with the functionality separately, we can bring the functionality into the content with Contextual Tools.

Contextual Tools are the Web's version of the desktop's right-click menus. Instead of having to right-click to reveal a menu, we can reveal tools in context with the content. We can do this in a number of ways:

Always-Visible Tools

Place Contextual Tools directly in the content.

Hover-Reveal Tools

Show Contextual Tools on mouse hover.

Toggle-Reveal Tools

A master switch to toggle on/off Contextual Tools for the page.

Multi-Level Tools

Progressively reveal actions based on user interaction.

Secondary Menus

Show a secondary menu (usually by right-clicking on an object).

6.Explain in detail about overlays.

Overlays are really just lightweight pop ups. We use the term *lightweight* to make a clear distinction between it and the normal idea of a *browser pop up*. Browser pop ups are created as a new browser window (Figure 5-1). *Lightweight overlays* are shown within the browser page as an overlay (Figure 5-2). Older style browser pop ups are undesirable because:

Browser pop ups display a new browser window. As a result • these windows often take time and a sizeable chunk of system resources to create. • Browser pop ups often display browser interface

controls (e.g., a URL bar). Due to security concerns, in Internet Explorer 7 the URL bar is a permanent fixture on any browser pop-up window.

Dialog Overlay

Dialog Overlays replace the old style browser pop ups. Netflix provides a clear example of a very simple Dialog Overlay. In the "previously viewed movies for sale" section, a user can click on a "Buy" button to purchase a DVD. Since the customer purchasing the DVD is a member of Netflix, all the pertinent shipping and purchasing information is already on record. The complete checkout experience can be provided in a single overlay.

Detail Overlay

The second type of overlay is somewhat new to web applications. The Detail Overlay allows an overlay to present additional information when the user clicks or hovers over a link or section of content. Toolkits now make it easier to create overlays across different browsers and to request additional information from the server without refreshing the page.

7. Explain in detail about inlays.

Dialog Inlay

A simple technique is to expand a part of the page, revealing a dialog area within the page. The BBC recently began experimenting with using a Dialog Inlay as a way to reveal customization controls for its home page.

Activation

Considerations

Of course an overlay could have been used instead. However, the problem with overlays is that no matter where they get placed, they will end up hiding information. Inlays get around this problem by inserting themselves directly into the context of the page.

List Inlay

Lists are a great place to use Inlays. Instead of requiring the user to navigate to a new page for an item's detail or popping up the information in an Overlay, the information can be shown with a List Inlay in context. The List Inlay works as an effective way to hide detail until needed—while at the same time preserving space on the page for high-level overview information. Google Reader provides an expanded view and a list view for unread blog articles.

Detail Inlay

A common idiom is to provide additional detail about items shown on a page. We saw this with the example of the Netflix movie detail pop up in Chapter 5 (Figure 5-8). Hovering over a movie revealed a Detail Overlay calling out the back-of-the-box information.

8.Explain in detail about virtual pages and its methods.

Patterns that support virtual pages include:

• Virtual Scrolling

• Inline Paging

• Scrolled Paging

• Panning

• Zoomable User Interface

Loading status

There are a few downsides to the Yahoo! Mail version of Virtual Scrolling. First, if the loading is slow, it spoils the illusion that the data is continuous. Second, since the scrollbar does not give any indication of where users are located in the data, they have to guess how far down to scroll. A remedy would be to apply a constantly updating status while the user is scrolling.

Progressive loading

Microsoft has applied Virtual Scrolling to its image search. However, it implements it in a different manner than Yahoo! Mail. Instead of all content being virtually loaded (and the scrollbar reflecting this), the scrollbar reflects what has been load

Inline Paging

What if instead of scrolling through content we just wanted to make pagination feel less like a page switch? By only switching the content in and leaving the rest of the page stable, we can create an Inline Paging experience.

9. Explain the steps followed in process flow.

Google Blogger

The popular site Google Blogger generally makes it easy to create and publish blogs. One thing it does not make easy, though, is deleting comments that others may leave on your blog. This is especially difficult when you are the victim of hundreds of spam comments left by nefarious companies hoping to increase their search ranking. Blogger forces you to delete these comments through a three-step process. Each step is an individual page, all punctuated with a page refresh (Figure 8-1).

Scroll to find the offending comment.
2. Click the trash icon to delete the comment.
3. After page refreshes, click the "Remove Forever" checkbox.
4. Click the "Delete Comment" button.
5. After the page refreshes, click the link to return to my blog article.
6. Repeat steps 1–5 for each article with spam comments.

The Magic Principle

Alan Cooper discusses a wonderful technique for getting away from a technology-driven approach and discovering the underlying mental model of the user. He calls it the "magic principle."

Ask the question, "What if when trying to complete a task the user could invoke some magic?" For example, let's look at the problem of taking and sharing photos. The process for this task breaks down like this:

• Take pictures with a digital camera.
• Sometime later, upload the photos to a photo site like Flickr. This involves:
— Finding the cable.
— Starting iTunes.
— Importing all photos.

Interactive Single-Page Process
• Inline Assistant Process
• Configurator Process
• Overlay Process
• Static Single-Page Process

Inline Assistant Process

Another common place where multiple pages are used to complete a process is when adding items to a shopping cart. As mentioned earlier, Amazon provides the typical experience. So what magic can we apply to move this from a multi-page experience to a single-page experience? Instead of thinking about the cart as a process, we can think about it as a real-world object. Given this mindset, the cart can be realized in the interface as an object and be made available on the page.

10. Explain case study topic: Look up patterns and its types.

A good portion of application interfaces is involved in looking up information. Whether an application is performing direct searches, filtering search results, or aiding in input, there are a lot of opportunities to provide lookup assistance.

direct selection method

Auto Complete

Auto Complete is a powerful pattern that benefits from a reactive interface. As the user types
input into a field, a drop-down menu of matching values is displayed. When done right, the choice that best matches will be auto-selected. The user can stop typing and accept the choice that has been matched or choose a different value from the list. The selected value is then entered into the field. Yahoo! Mail uses Auto Complete for email addresses.

Live Suggest

A very close cousin to Auto Complete is the Live Suggest pattern (also known as winnowing).While Auto Complete provides suggested real-time values for an input field, Live Suggest provides real-time search term suggestions for creating a search. The context switch from input field to search box and from input value to term suggestion puts a different twist on the interaction.

Live Search

Yet another close relative to both Auto Complete and Live Suggest is Live Search. As with Live Suggest, a search query is being formulated. But instead of displaying suggested search terms, actual live search results are shown in real time. A clear example of this is ZUGGEST for Amazon. Not affiliated with Amazon.com, Francis Shanahan created a Live Search for Amazon products.