MODULE 2

2.1 OBSTACLES AND PITFALLS IN THE DEVELOPMENT PATH

- The development path of a computer system is littered with obstacles and traps, many of them human in nature.
- Gould (1988) has made these general observations about design:
  - Nobody ever gets it right the first time.
  - Development is chock-full of surprises.
  - Good design requires living in a sea of changes.
  - Making contracts to ignore change will never eliminate the need for change.
  - Even if you have made the best system humanly possible, people will still make mistakes when using it.
  - Designers need good tools.
- The first five conditions listed will occur naturally because people are people, both as users and as developers. These kinds of behavior must be understood and accepted in design.
- User mistakes, while they will always occur, can be reduced. Guidelines in the various design steps address this problem.
- Pitfalls in the design process exist because of a flawed design process, including a failure to address critical design issues, an improper focus of attention, or development team organization failures.
- Common pitfalls are:
  - No early analysis and understanding of the user's needs and expectations.
  - A focus on using design features or components that are "neat" or "glitzy."
  - Little or no creation of design element prototypes.
  - No usability testing.
  - No common design team vision of user interface design goals.
  - Poor communication between members of the development team.

2.1.1 Designing for People: The Five Commandments

The complexity of a graphical or Web interface will always magnify any problems that do occur. While obstacles to design will always exist, pitfalls can be eliminated if the following design commandments remain foremost in the designer's mind.

1. **Gain a complete understanding of users and their tasks.** The users are the customers. Today, people expect a level of design sophistication from all interfaces, including Web sites. The product, system or Web site must be geared to people's needs, not those of the developers.

A wide gap in technical abilities, goals, and attitudes often exists between users and developers. A failure to understand the differences will doom a product or system to failure.

2. **Solicit early and ongoing user involvement.** Involving the users in design from the beginning provides a direct conduit to the knowledge they possess about jobs, tasks, and needs. Involvement also allows the developer to confront a person's resistance to change. People dislike change for a variety of reasons like fear of the unknown and lack of identification with the system. Involvement in design removes the unknown and gives the user a stake in the system or identification with it. User involvement should be based on job or task knowledge, not status or position. The boss seldom knows what is really happening out in the office.
3. **Perform rapid prototyping and testing.** Prototyping and testing the product will quickly identify problems and allow you to develop solutions. The design process is complex and human behavior is still not well understood. Prototyping and testing must be continually performed during all stages of development to uncover all potential defects. If thorough testing is not performed before product release, the testing will occur in the user's office. Encountering a series of problems early in system use will create a negative first impression in the customer's mind, and this may harden quickly, creating attitudes that may be difficult to change. It is also much harder and costlier to fix a product after its release. In many instances, people may adapt to, or become dependent upon, a design, even if it is inefficient. This also makes future modifications much more difficult.
4. **Modify and iterate the design as much as necessary.** Design proceeds through a series of stages. Problems detected in one stage may force the developer to revisit a previous stage. This is normal and should be expected. Establish user performance and acceptance criteria and continue testing and modifying until all design goals are met.
5. **Integrate the design of *all* the system components.** The software, the documentation, the help function, and training needs are all important elements of a graphical system or Web site and all should be developed concurrently. A system is being constructed, not simply software. Concurrent development of all pieces will point out possible problems earlier in the design process, allowing them to be more effectively addressed. Time will also exist for design trade-offs to be thought out more carefully.

2.2 USABILITY

Bennett (1979) was the first to use the term *usability* to describe the effectiveness of human performance. In the following years a more formal definition was proposed by Shackel (1981)

and modified by Bennett (1984). Finally, Shackel (1991) defined usability as "the capability to be used by humans easily and effectively, where, easily = to a specified level of subjective assessment, effectively = to a specified level of human performance."

2.2.1 Usability Assessment in the Design Process

- Usability assessment should begin in the early stages of the product development cycle and should be continually applied throughout the process.
- Assessment should include user's entire experience, and all the product's important components.

2.2.2 Common Usability Problems

- Mandel (1994) lists the 10 most common usability problems in graphical systems as reported by IBM usability specialists. They are:
  1. Ambiguous menus and icons.
  2. Languages that permit only single-direction movement through a system.
  3. Input and direct manipulation limits.
  4. Highlighting and selection limitations.
  5. Unclear step sequences.
  6. More steps to manage the interface than to perform tasks.
  7. Complex linkage between and within applications.
  8. Inadequate feedback and confirmation.
  9. Lack of system anticipation and intelligence.
  10. Inadequate error messages, help, tutorials, and documentation.

- The Web, with its dynamic capabilities has unleashed what seems like more than its own share of usability problems.
- Web usability characteristics particularly wasteful of people's time, and often quite irritating, are:
  1. **Visual clutter.** A lack of "white space," meaningless graphics, and unnecessary and wasteful decoration often turn pages into jungles of visual noise. Meaningful content lies hidden within the unending forest of vines and trees, forcing the user to waste countless minutes searching for what is relevant. Useless displayed elements are actually a form of visual noise.
  2. **Impaired information readability.** Page readability is diminished by poor developer choices in typefaces, colors, and graphics. Use of innumerable typefaces and

kaleidoscopic colors wrestle meaning from the screen. A person's attention is directed towards trying to understand why the differences exist, instead of being focused toward identifying and understanding the page's content. Backgrounds that are brightly colored or contain pictures or patterns greatly diminish the legibility of the overwritten text.

3. **Incomprehensible components.** Some design elements do not give the user any clue about their function. Some icons and graphics do not contain text to explain what they do. Some buttons don't look at all like command buttons, forcing the user to "minesweep" the screen with a mouse to locate the objects that can be used to do something. Command buttons or areas that give no visual indication that they are clickable often won't be clicked. Language is also often confusing, with the developer's terminology being used, not that of the user.
4. **Annoying distractions.** Elements constantly in motion, scrolling marquees or text, blinking text, or looping continually running animations compete with meaningful content for the user's eye's and attention—and destroy a page's readability. Automatically presented music or other sounds interrupt one's concentration. Non-requested pop-up widows must be removed that waste more of the user's time. A person's senses are under constant attack, and the benefits afforded by one's peripheral vision are negated.
5. **Confusing navigation.** A site's structure often resembles a maze of twisting pages into which the user wanders and is quite soon lost. Poor, little, or no organization exists among pages. The size and depth of many Web sites can eventually lead to a "lost in space" feeling as perceived site structure evaporates as one navigates. Embarking on a side trip can lead to a radical change in context or a path with no signposts or landmarks. Navigation links lead to dead-ends from which there is no return, or boomerang you right back to the spot where you are standing without you being aware of it. Some navigation elements are invisible. Confusing navigation violates expectations and results in disturbing unexpected behavior.
6. **Inefficient navigation.** A person must transverse content-free pages to find what is meaningful. One whole screen is used to point to another. Large graphics waste screen space and add to the page count. The path through the navigation maze is often long and tedious. Reams of useless data must be sifted through before a need can be fulfilled. Massive use of short pages with little content often creates the feeling that one is "link drunk."
7. **Inefficient operations.** Time is wasted doing many things. Page download times can be excessive. Pages that contain, for example, large graphics and maps, large chunky

headings, or many colors, take longer to download than text. Excessive information fragmentation can require navigation of long chains of links to reach relevant material, also accelerating user disorientation.

8. **Excessive or inefficient page scrolling.** Long pages requiring scrolling frequently lead to the user's losing context as related information's spatial proximity increases and some information entirely disappears from view and, therefore, from memory. Out of sight is often out of mind. If navigation elements and important content are hidden below the page top, they may be missed entirely. To have to scroll to do something important or complete a task can be very annoying.
9. **Information overload.** Poorly organized or large amounts of information taxes one's memory and can be overwhelming. Heavy mental loads can result from making decisions concerning which links to follow and which to abandon, given the large number of choices available. One easily becomes buried in decisions and information. Requiring even minimal amounts of learning to use a Web site adds to the mental load.
10. **Design inconsistency.** The business system user may visit a handful of systems in one day, the Web user may visit dozens, or many more. It is expected that site differences will and must exist because each Web site owner strives for its own identity. For the user's sake, some consistency must exist to permit a seamless flow between sites. Consistency is needed in, for example, navigation element location on a page and the look of navigation buttons (raised). When users are forced to remember different color meanings in different places, it causes confusion between links and underlined text. Examples of inappropriate uses abound in design - use of check boxes instead of radio buttons for mutually exclusive options, use of drop-down list boxes instead of combination boxes when the task mostly requires keyboard form fill-in. The Web is a form of the graphical user interface, and GUI guidelines should be followed.
11. **Outdated information.** A Web site should be "current." Outdated information destroys a site's credibility in the minds of many users, and therefore its usefulness. A useless site is not very usable.

12. **Stale design caused by emulation of printed documents and past systems.** The Web is a new medium with expanded user interaction and information display possibilities. Web sites should be rethought and redesigned using the most appropriate and robust design techniques available. Developers often have created a product to please themselves and "look cool," not to please their users. Symptoms of this approach include overuse of

bleeding edge technology, a focus on sparkle, and jumping to implement the latest Internet technique or buzzword.

2.2.3 Some Practical Measures of Usability

Some usability problems are:

1) **Are people asking a lot of questions or often reaching for a manual?** Many questions or frequent glances at manuals are signs that things are not as clear and intuitive as they should be. When in doubt, the first reaction of many people is to ask someone for assistance. When no one is around, then we look in a manual
2) **Are frequent exasperation responses heard?** "Oh damn!" or similar reactions are usually used to express annoyance or frustration. Their frequency, and loudness, may foretell a strong rejection of a product. The absence of exasperation may not represent acceptance. Some people are not as expressive in their language, or are better able to smother their feelings.
3) **Are there many irrelevant actions being performed?** Are people doing things the hard way? Are there incidental actions required for, but not directly related to, doing a job? These include excessive mouse clicks or keyboard strokes to accomplish something, or going through many operations to find the right page in a manual or the right window or page in the display.
4) **Are there many things to ignore?** Are there many elements on the screen that the user must disregard? Are there many "doesn't pertain to me" items? If so, remember, they still consume a portion of a person's visual or information processing capacities, detracting from the capacities a person could devote to relevant things.
5) **Do a number of people want to use the product?** None of us goes out of our way to make our own lives more difficult. (Unfortunately, other people may, however.) We tend to gravitate to things easy to work with or do. If a lot of people want to use it, it probably has a higher usability score. Attitudes may be a very powerful factor in a system's or Web site's acceptance.

2.2.4 Some Objective Measures of Usability

- Shackel (1991) presents the following more objective criteria for measuring usability.
  - How *effective* is the interface? Can the required range of tasks be accomplished?:
    - At better than some required level of performance (for example, in terms of speed and errors)?
    - By some required percentage of the specified target range of users?

- Within some required proportion of the range of usage environments?

- How *learnable* is the interface? Can the interface be learned?
  - Within some specified time from commissioning and start of user training?
  - Based on some specified amount of training and user support?
  - Within some specified relearning time each time for intermittent users?
- How *flexible* is the interface? Is it flexible enough to:
  - Allow some specified percentage variation in tasks and/or environments beyond those first specified?

- What are the *attitudes* of the users? Are they:
  - Within acceptable levels of human cost in terms of tiredness, discomfort, frustration, and personal effort?
  - Such that satisfaction causes continued and enhanced usage of the system?

❖ Human performance goals in system use should be stated in quantitative and measurable ways.

❖ Without performance goals you will never know if you have achieved them, or how successful the system really is.

❖ Clear and concrete goals also provide objectives for usability testing and ensure that a faulty or unsatisfactory product will not be released.

❖ Values for the various criteria should be specified in absolute terms. An absolute goal might be "Task A must be performed by a first-time user in 12 minutes with no errors with 30 minutes of training and without referring to a manual."

❖ Goals may also be set in relative terms. For example, "Task B must be performed 50 percent faster than it was using the previous system."

❖ The level of established goals will depend on the capabilities of the user, the capabilities of the system, and the objectives of the system. In addition to providing commitments to a certain level of quality, goals become the foundation for the system test plan.

2.3 IMPORTANT HUMAN CHARACTERISTICS IN DESIGN

1. Perception

❖ Perception is our awareness and understanding of the elements and objects of our environment through the physical sensation of our various senses, including sight, sound, smell, and so forth.

❖ Perception is influenced partly by *experience*.

❖ Other perceptual characteristics include the following:

- **Proximity**. Our eyes and mind see objects as belonging together if they are near each other in space.
- **Similarity**. Our eyes and mind see objects as belonging together if they share a common visual property, such as color, size, shape, brightness, or orientation.
- **Matching patterns.** We respond similarly to the same shape in different sizes. The letters of the alphabet, for example, possess the same meaning, regardless of physical size.
- **Succinctness.** We see an object as having some perfect or simple shape because perfection or simplicity is easier to remember.
- **Closure.** Our perception is synthetic; it establishes meaningful wholes. If something does not quite close itself, such as a circle, square, triangle, or word, we see it as closed anyway.
- **Unity.** Objects that form closed shapes are perceived as a group.
- **Continuity**. Shortened lines may be automatically extended.
- **Balance.** We desire stabilization or equilibrium in our viewing environment. Vertical, horizontal, and right angles are the most visually satisfying and easiest to look at in our viewing environment.
- **Expectancies.** Sometimes we perceive not what is there but what we expect to be there. Missing a spelling mistake in proofreading something we write is often an example of a perceptual expectancy error; we see not how a word *is* spelled, but how we *expect* to see it spelled.
- **Context.** Context, environment, and surroundings also influence individual perception. For example, two drawn lines of the same length may look the same length or different lengths, depending on the angle of adjacent lines or what other people have said about the size of the lines.
- **Signals versus noise.** Our sensing mechanisms are bombarded by many stimuli, some of which are important and some of which are not. Important stimuli are called signals; those that are not important or unwanted are called noise. Signals are more quickly comprehended if they are easily distinguishable from noise in our sensory environment. Noise interferes with the perception of signals to the extent that they are similar to one another. Noise can even mask a critical signal.

  The elements of a screen assume the quality of signal or noise, depending on the actions and thought processes of the user. Once a screen is first presented and has to be identified as being the correct one, the screen's title may be the signal, the other elements it contains simply being noise. When the screen is being used, the data it contains becomes the signal,

and the title now reverts to noise. Other elements of the screen rise and fall in importance, assuming the roles of either signals or noise, depending on the user's needs of the moment. The goal in design is to allow screen elements to easily assume the quality of signal or noise, as the needs and tasks of the user change from moment to moment.

2. Memory

- Memory is viewed as consisting of two components, long-term and short-term (or working) memory.
- *Short-term*, or working, memory receives information from either the senses or longterm memory, but usually cannot receive both at once, the senses being processed separately.
- Within short-term memory a limited amount of information processing takes place. Information stored within it is variously thought to last from 10 to 30 seconds.
- Knowledge, experience, and familiarity govern the size and complexity of the information that can be remembered.
- *Long-term* memory contains the knowledge we possess. Information received in short-term memory is transferred to it and encoded within it, a process called learning.
- Learning process improved if the information being transferred from short-term memory has structure and is meaningful and familiar and through repetition.
- Unlike short-term memory, with its distinct limitations, long-term memory capacity is thought to be unlimited.
- To enhance system usability: Present information in an organized, structured, familiar, and meaningful way. Place all required information for task performance in close physical proximity. Give the user control over the pace of information presentation.

3. Sensory Storage

- Sensory storage is the buffer where the automatic processing of information collected from our senses takes place.
- It is an unconscious process, large, attentive to the environment, quick to detect changes, and constantly being replaced by newly gathered stimuli.
- It constantly scans the environment for things that are important to pass on to higher memory.
- Repeated and excessive stimulation can fatigue the sensory storage mechanism, making it less attentive and unable to distinguish what is important (called *habituation*). Avoid unnecessarily stressing it.
- Design the interface so that all aspects and elements serve a definite purpose. Eliminating interface noise will ensure that important things will be less likely to be missed.

4. Visual Acuity

- Capacity of the eye to resolve details is called *visual acuity*.
- phenomenon that results in an object becoming more distinct as we turn our eyes towards it and rapidly losing distinctness as we turn our eyes away- visual angle from point of fixation increases.
- The eye is also never perfectly steady as it sees; it trembles slightly. This tremor improves the detection of edges of objects being looked at, thus improving acuity. Patterns of closely spaced lines or dots are seen to shimmer due to this tremor. This movement can be distracting and disturbing. Patterns for fill-in areas of screens (bars, circles, and so on.) must be carefully chosen to avoid visual distraction.

5. Foveal and Peripheral Vision

- *Foveal vision* is used to focus directly on something.
- *peripheral vision* senses anything in the area surrounding the location we are looking at, but what is there cannot be clearly resolved because of the limitations in visual acuity just described.
- Foveal and peripheral vision maintain, at the same time, a cooperative and a competitive relationship.
- Peripheral vision can aid a visual search, but can also be distracting. In its cooperative nature, peripheral vision is thought to provide clues to where the eye should go next in the visual search of a screen. Patterns, shapes, and alignments peripherally visible can guide the eye in a systematic way through a screen.
- In its competitive nature, peripheral vision can compete with foveal vision for attention. What is sensed in the periphery is passed on to our information-processing system along with what is actively being viewed foveally.

6. Information Processing

- The information that our senses collect has to be processed in some meaningful way.
- Two levels of information processing
  - the highest level, is identified with consciousness and working memory. It is limited, slow, and sequential, and is used for reading and understanding.
  - there exists a lower level of information processing, and the limit of its capacity is unknown. This lower level processes familiar information rapidly, in parallel with the higher level, and without conscious effort.
- Repetition and learning results in a shift of control from the higher level to the lower level.

- Both levels function simultaneously, the higher level performing reasoning and problem solving, the lower level perceiving the physical form of information sensed.
- Visual distinctiveness of a screen is a strong contributor to lower level information processing. If a screen is jammed with information and cluttered, it loses its uniqueness and can only be identified through the more time-consuming, and thought-interrupting, reading process.

7. Mental Models

- As a result of our experiences and culture, we develop mental models of things and people we interact with.
- A mental model is an internal representation of a person's current understanding of something.
- Mental models are gradually developed in order to understand something, explain things, make decisions, do something, or interact with another person. They also enable a person to predict the actions necessary to do things if the action has been forgotten or has not yet been encountered.
- When confronting a new computer system, people will bring their own expectations and preconceptions based upon mental models they have formed doing things in their daily life. If the system conforms to the mental models a person has developed, the model is reinforced and the system's use feels more "intuitive." If not, difficulties in learning to use the system will be encountered. This is why in design it is critical that a user's mental models be to identified and understood.
- The key to forming a transferable mental model of a system is design consistency and design standards.

8. Movement Control

- Once data has been perceived and an appropriate action decided upon, a response must be made.
- in many cases the response is a movement.
- In computer systems, movements include such activities as pressing keyboard keys, moving the screen pointer by pushing a mouse or rotating a trackball, or clicking a mouse button.
- Fitts' Law for screen design states that:

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

This simply means that the bigger the target is, or the closer the target is, the faster it will be reached.

- The implications in screen design are:

  Provide large objects for important functions.

  Take advantage of the "pinning" actions of the sides, top, bottom, and corners of the screen.

- Big buttons are better than small buttons. They provide a larger target for the user to access with the screen pointer.
- Create toolbar icons that "bleed" into the edges of a display, rather than those leaving a one-pixel non-clickable edge along the display boundary. The edge of the screen will stop or "pin" the pointer's movement at a position over toolbar, permitting much faster movement to the toolbar.
- A one-pixel edge will require more careful positioning of the pointer over the toolbar.

9. Learning

- Learning is the process of encoding in long-term memory information that is contained in short-term memory.
- Our ability to learn clearly differentiates people from machines.
- Given enough time people can improve their performance in almost any task. Designers use our learning ability as an excuse to justify complex design.
- A design developed to minimize human learning time can greatly accelerate human performance. People prefer to stick with what they know. Unproductive time spent learning is something frequently avoided.
- People are very sensitive to even minor changes in the user interface, and that such changes may lead to problems in transferring from one system to another.
- The "perception" of having to learn huge amounts of information is enough to keep some people from even using a system.
- Learning can be enhanced if it:
  - Allows skills acquired in one situation to be used in another somewhat like it. Design consistency accomplishes this.
  - Provides complete and prompt feedback.
  - Is phased, that is, it requires a person to know only the information needed at that stage of the learning process.

10. Skill

- goal of human performance is to perform skillfully.
- To do so requires linking inputs and responses into a sequence of action.
- The essence of skill is performance of actions or movements in the correct time sequence with adequate precision. It is characterized by consistency and economy of effort.

- Economy of effort is achieved by establishing a work pace that represents optimum efficiency. It is accomplished by increasing mastery of the system through such things as progressive learning of shortcuts, increased speed, and easier access to information or data.
- Skills are hierarchical in nature, and many basic skills may be integrated to form increasingly complex ones.
- System and screen design must permit development of increasingly skillful performance.

11. Individual Differences

- An advantageous human characteristic is that we all differ in looks, feelings, motor abilities, intellectual abilities, learning abilities and speed, and so on.
- In a keyboard data entry task, for example, the best typists will probably be twice as fast as the poorest and make 10 times fewer errors.
- Individual differences complicate design because the design must permit people with widely varying characteristics to satisfactorily and comfortably learn the task or job, or use the Web site.
- Technology offers the possibility of tailoring jobs to the specific needs of people with varying and changing learning or skill levels. Multiple versions of a system can easily be created.
- Design must provide for the needs of all potential users.

2.4 HUMAN INTERACTION SPEEDS

- The speed at which people can perform using various communication methods has been studied by a number of researchers.
- The following, as summarized by Bailey (2000), have been found to be typical interaction speeds for various tasks.
  - **Reading.** The average adult, reading English prose in the United States, has a reading speed in the order of 250–300 words per minute. Proofreading text on paper has been found to occur at about 200 words per minute, on a computer monitor, about 180 words per minute. Nontraditional reading method that has dramatically increased reading speeds is called Rapid Serial Visual Presentation, or RSVP. In this technique single words are presented one at a time in the center of a screen. New words continually replace old words at a rate set by the reader.
  - **Listening.** Words can be comfortably heard and understood at a rate of 150 to 160 words per minute. This is generally the recommended rate for audio books and video narration It is found that when normal speech is speeded up using compression, a speed of 210 words per minute results in no loss of comprehension.

- **Speaking.** Dictating to a computer occurs at a rate of about 105 words per minute. Speech recognizer misrecognitions often occur, however, and when word correction times are factored in, the speed drops significantly, to an average of 25 words per minute. It was also found that the speaking rate of new users was 14 words per minute during transcription and 8 words per minute during composition.
- **Keying.** Fast typewriter typists can key at rates of 150 words per minute and higher. Average typing speed is considered to be about 60–70 words per minute. Computer keying has been found to be much slower, however. Speed for simple transcription was only 33 words per minute and for composition only 19 words per minute. Two-finger typists can key memorized text at 37 words per minute and copied text at 27 words per minute.
- **Hand printing.** People hand print memorized text at about 31 words per minute. Text is copied at about 22 words per minute.

2.5 BUSINESS DEFINITION AND REQUIREMENTS ANALYSIS

- objective of this phase to establish the need for a system.
- A requirement is an objective that must be met.
- A product description is developed and refined, based on input from users or marketing. There are many techniques for capturing information for determining requirements.
- Techniques are classified as direct and indirect.
- Direct methods consist of face-to-face meetings with, or actual viewing of, users to solicit requirements.
- Indirect methods impose an intermediary, someone or something, between the users and the developers.
- Before beginning the analysis, the developer should be aware of the policies and work culture of the organization being studied. He or she should also be familiar with any current system or process the new system is intended to supplement or replace.

2.5.1 Direct Methods

- Advantage: they provide to hear the user's comments in person and firsthand. Person-to-person encounters permit multiple channels of communication (body language, voice inflections, and so on) and provide the opportunity to immediately follow up on vague or incomplete data.

1. ***Individual Face-to-Face Interview***
   - A one-on-one visit is held with the user. It may be structured or more open-ended.
   - The interview must have focus. Topics to be covered must be carefully planned so data is collected in a common framework. Ensure that all important aspects are thoroughly covered.

- Formal questionnaire should not be used.
- Useful topics to ask the user to describe in an interview include:
  - ✓ activities performed in completing a task or achieving a goal or objective.
  - ✓ methods used to perform an activity.
  - ✓ interactions that exist with other people or systems.
- It is also very useful to also uncover any:
  - ✓ Potential measures of system usability
  - ✓ Unmentioned exceptions to standard policies or procedures.
  - ✓ Relevant knowledge the user must possess to perform the activity.
- If designing a Web site, the following kinds of interview questions are appropriate for asking potential users:
  - ✓ Present a site outline or proposal and then solicit comments on the thoroughness of content coverage, and suggestions for additional content.
  - ✓ Ask users to describe situations in which the proposed Web site might be useful.
  - ✓ Ask users to describe what is liked and disliked about the Web sites of potential competitors.
  - ✓ Ask users to describe how particular Web site tasks should be accomplished.
- Time must also be allowed for free conversation in interviews.
- Recording the session for playback to the entire design team provides all involved with some insights into user needs.
- Advantages of a personal interview: - you can give the user your full attention,
  - can easily include follow-up questions to gain additional information
  - will have more time to discuss topics in detail
  - will derive a deeper understanding of your users, their experiences, attitudes, beliefs, and desires.
- Disadvantages of interviews: - they can be costly and time-consuming to conduct
  - someone skilled in interviewing techniques should perform them.
- The interviewer must establish a positive relationship with the user, ask questions in a neutral manner, be a good listener, and know when and how to probe for more information.

2. ***Telephone Interview or Survey***
   - interview conducted using the telephone.
   - It must have structure and be well planned. Arranging the interview in advance allows the user to prepare for it.

- Telephone interviews less expensive and less invasive than personal interviews.
- can be used much more frequently and are extremely effective for very specific information.
- Disadvantages: - impossible to gather contextual information, such as a description of the working environment
    - replies may be easily influenced by the interviewer's comments, and body language cues are missing.
    - it may be difficult to contact the right person for the telephone interview.

3. ***Traditional Focus Group***
    - Small group of users (8 to 12) and a moderator brought together to discuss the requirements.
    - discussion is loosely structured. Range of topics must be determined beforehand.
    - typical session lasts about two hours.
    - purpose of a focus group - probe user's experiences, attitudes, beliefs, and desires, and obtain their reactions to ideas or prototypes.
    - Focus groups not usually useful for establishing how users really work or what kinds of usability problems they really have.
    - Focus group discussion can be influenced by group dynamics, for good or bad.
    - Recording of the session, either video or audio, will permit later detailed analysis of participants comments. It can also be played for the entire design team, providing insights into user needs for all developers.
    - Setting up focus group involves the following:
        - ✓ Establish the objectives of the session.
        - ✓ Select participants representing typical users, or potential users.
        - ✓ Write a script for the moderator to follow.
        - ✓ Find a skilled moderator to facilitate discussion, to ensure that the discussion remains focused on relevant topics, and to ensure that everyone participates.
        - ✓ Allow the moderator flexibility in using the script.
        - ✓ Take good notes, using the session recording for backup and clarification.

4. ***Facilitated Team Workshop***
    - similar in structure and content to a traditional focus group but is slightly less formal.
    - Team workshops have had the potential to provide much useful information. They do require a great deal of time to organize and run.

5. ***Observational Field Study***

- To see and learn what users actually do, they are watched and followed in their own environment, office, or home, in a range of contexts for a period of time.
- Observation provides good insight into tasks being performed, the working environment and conditions, the social environment, and working practices.
- more objective, natural, and realistic. Observation can be time-consuming and expensive.
- Video recording of the observation sessions will permit detailed task analysis. Playing the recording for the entire design team again provides all involved with some insights into user tasks.

6. ***Requirements Prototyping***

   A demonstration model, or very early prototype, is presented to users for their comments concerning functionality.

7. ***User-Interface Prototyping***

   A demonstration model, or early prototype, is presented to users to uncover user interface issues and problems.

8. ***Usability Laboratory Testing***
   - A special laboratory is constructed and users brought in to perform actual newly designed tasks.
   - They are observed and results measured, and evaluated to establish the usability of the product at that point in time.
   - Usability tests uncover what people actually do, not what they think they do, a common problem with verbal descriptions.
   - same scenarios can be presented to multiple users, providing comparative data from several users.
   - Problems uncovered may result in modification of the requirements.
   - Usability labs can generate much useful information but are expensive to create and operate.

9. ***Card Sorting for Web Sites***
   - technique used to establish hierarchical groupings of information for Web sites.
   - used only after gathering substantial site content information using other analysis techniques.
   - Potential content topics are placed on individual index cards and users are asked to sort the cards into groupings that are meaningful to them.
   - Card sorting assists in building the site's structure, map, and page content.
   - process is as follows:

- ✓ From previous analyses, identify about 50 content topics and inscribe them on index cards. Limit topics to no more than 100.
- ✓ Provide blank index cards for names of additional topics the participant may want to add, and colored blank cards for groupings that the participant will be asked to create.
- ✓ Number the cards on the back.
- ✓ Arrange for a facility with large enough table for spreading out cards.
- ✓ Select participants representing a range of users. Use one or two people at a time and 5 to 12 in total.
- ✓ Explain the process to the participants, saying that you are trying to determine what categories of information will be useful, what groupings make sense, and what the groupings should be called.
- ✓ Ask the participants to sort the cards and talk out loud while doing so. Advise the participants that additional content cards may be named and added as they think necessary during the sorting process.
- ✓ Observe and take notes as the participants talk about what they are doing. Pay particular attention to the sorting rationale.
- ✓ Upon finishing the sorting, if a participant has too many groupings ask that they be arranged hierarchically.
- ✓ Ask participants to provide a name for each grouping on the colored blank cards, using words that the user would expect to see that would lead them to that particular grouping.
- ✓ Make a record of the groupings using the numbers on the back of each card.
- ✓ Reshuffle the cards for the next session.
- ✓ When finished, analyze the results looking for commonalities among the different sorting sessions.

- ➢ The sorting can also be accomplished on the Web. The National Institute of Standards and Technology (NIST, 2001) has developed a card-sorting tool.
- ➢ The designer sets up the cards and names the categories. The user then sorts by dragging and dropping.

2.5.2 Indirect Methods

- ❖ An indirect method of requirements determination places an intermediary between the developer and the user.
- ❖ This intermediary may be electronic or another person.

- Working through an intermediary takes away the multichannel communication advantages of face-to-face user-developer contact.
- Imposition of a human intermediary can also create additional problems.
  - ✓ there may be a filtering or distortion of the message, either intentional or unintentional.
  - ✓ the intermediary may not possess a complete, or current, understanding of user's needs, passing on an incomplete or incorrect message.
  - ✓ the intermediary may be a mechanism that discourages direct user-developer contact for political reasons.
- Indirect methods include the following.

1. ***MIS Intermediary***
   - A company representative who defines the user's goals and needs to designers and developers fulfills this intermediary role.
   - representative may come from the Information Services department itself, or he or she may be from the using department.
   - too often this person does not have the breadth of knowledge needed to satisfy all design requirements.

2. ***Paper Survey or Questionnaire***
   - A paper questionnaire or survey administered to a sample of users to obtain their needs.
   - Questionnaires have the potential to be used for a large target audience located most anywhere, and are much cheaper than customer visits.
   - Generally, have a low return rate, often generating responses only from those "very happy" or "very unhappy."
   - They may take a long time to collect and may be difficult to analyze.
   - Questionnaires useful for determining a user's attitudes, experiences and desires, but not for determining actual tasks and behaviors.
   - Questionnaires should be composed mostly of closed questions (yes/no, multiple choice, short answer, and so on).
   - Open-ended questions require much more analysis.
   - Questionnaires should be relatively short and created by someone experienced in their design.

3. ***Electronic Survey or Questionnaire***
   - A questionnaire or survey is administered to a sample of users via e-mail or the Web.
   - significantly less expensive than mailed surveys.

- speed of their return can also be much faster than those distributed in a paper format.
- In creating an electronic survey:
  - ✓ Determine the survey objectives.
  - ✓ Determine where you will find the people to complete the survey.
  - ✓ Create a mix of multiple choice and open-ended questions requiring short answers addressing the survey objectives.
  - ✓ Keep it short, about 10 items or less is preferable.
  - ✓ Keep it simple, requiring no more than 5–10 minutes to complete.
- consider a follow-up more detailed survey, or surveys, called *iterative surveys*.
- Ask people who complete and return the initial survey if they are willing to answer more detailed questions. Create and send the more detailed survey. It can address questions the initial survey raises.
- follow-up survey goal - ask the participant to prioritize their needs and to rank expected user tasks according to their importance.
- A third follow-up survey - designed to gather additional information about the most important requirements and tasks.
- Iterative surveys take a longer time to complete. thank participants for their help and time.

4. ***Electronic Focus Group Similar***

- An electronic focus group similar to a traditional focus group except but the discussion is accomplished electronically using specialized software on a workstation, e-mail, or a Web site.
- opportunity to immediately follow up on vague or incomplete data exists.
- All comments, ideas, and suggestions are available in hard-copy form for easier analysis. Specialized software can provide ratings or rankings of items presented in lists, a task requiring much more effort in a traditional focus group.
- advantages of electronic focus groups over traditional focus groups:
  - ✓ the discussion is less influenced by group dynamics.
  - ✓ has a smaller chance of being dominated by one or a few participants.
  - ✓ can be anonymous, leading to more honest comments and less caution in proposing new ideas
  - ✓ can generate more ideas in a shorter time since all participants can communicate at once
  - ✓ can lead to longer sessions since the participant is in a more comfortable "home environment" and not confined to a conference room.

- Disadvantages: depth and richness of verbal discussions does not exist and the communication enhancement aspects of seeing participant's body language are missing.

5. ***Marketing and Sales***
   - Company representatives regularly meet customers obtain suggestions or needs, current and potential.
   - information collected inexpensively, since the representative visits the company anyway.
   - Business representatives have knowledge of the nature of customers, the business, and the needs that have to be met.
   - dangers:
     - ✓ information may be collected from the wrong people
     - ✓ the representative may unintentionally bias questions
     - ✓ there may be many company "filters" between the representative's contact and the end users
     - ✓ quantities may sometimes be exaggerated.

6. ***Support Line***
   - Information collected by unit that helps customers with day-to-day problems (Customer Support, Technical Support, Help Desk, and so on).
   - fairly inexpensive and the target user audience is correct.
   - focus of this method is usually only on problems.

7. ***E-Mail, Bulletin Boards or Guest Book***
   - Problems, questions, and suggestions by users posted to a bulletin board, a guest book, or through e-mail are gathered and evaluated.
   - focus of this method usually only on problems.
   - responsibility is on the user to generate the recommendations, but this population often includes unhappy users.
   - fairly inexpensive.

8. ***User Group***
   - Improvements suggested by customer groups who convene periodically to discuss system and software usage are evaluated.
   - User groups have the potential to provide a lot of good information, if organized properly.
   - They require careful planning.

9. ***Competitor Analysis***

- Reviews of competitor's products, or Web sites, used to gather ideas, uncover design requirements, and identify tasks.
- designers can perform this evaluation or, users can be asked to perform the evaluation.

10. ***Trade Show***

- Customers at a trade show can be exposed to a mock-up or prototype and asked for comments.
- method dependent on the knowledge level of the customers and may provide only a superficial view of most prominent features.

11. ***Other Media Analysis***

- Analyze how other media, print or broadcast, present the process, information, or subject matter of interest.
- Findings used to gather ideas, uncover design requirements, and identify better ways to accomplish or show something.

12. ***System Testing***

- New requirements and feedback stemming from ongoing system testing can be accumulated, evaluated, and implemented as necessary.

2.6 REQUIREMENTS COLLECTION GUIDELINES

- Keil and Carmel evaluated the suitability and effectiveness of various requirements- gathering methods by collecting data on 28 projects in 17 different companies.
- Fourteen of the projects were rated as relatively successful, 14 as relatively unsuccessful.
- Each requirements collection method was defined as a developer-user *link*.
- Their findings and conclusions:

1. Establish 4 to 6 Different Developer-User Links

- more successful projects utilized greater number of developer-user links than the less successful projects.
- mean number of links for
  - ✓ the successful projects: 5.4
  - ✓ the less successful: 3.2.
- difference was statistically significant.
- Few projects used more than 60 percent of all possible links.
- Keil and Carmel recommend, based upon their data, that, at minimum, four different developer-user links must be established in the requirements-gathering process.

- They also concluded that the law of diminishing returns begins to set in after six links.
- Effectiveness ratings of the most commonly used links in their study were also obtained. On a 1 to 5 scale (1 = ineffective, 5 = very effective) the following methods had the highest ratings:
  - Custom projects (software developed for internal use and usually not for sale):

Facilitated Teams	5.0
User-Interface Prototype	4.0
Requirements Prototype	3.6
Interviews	3.5

  - Package projects (software developed for external use and usually for sale):

Support Line	4.3
Interviews	3.8
User-Interface Prototype	3.3
User Group	3.3

2. Provide the Most Reliance on Direct Links

- problems associated with the less successful projects resulted, at least in part, from too much reliance on indirect links, or using intermediaries.
- Ten of the 14 less successful projects had used none, or only one, *direct* link.
- The methods with the highest effectiveness ratings listed above were mostly direct links.
- Keil and Carmel caution that number of links is only a partial measure of user participation. How well the link is employed in practice is also very important.

2.7 DETERMINING BASIC BUSINESS FUNCTIONS

- Prepare a detailed description of what the product will do.
- Major system functions are listed and described, including critical system inputs and outputs.
- A flowchart of major functions is developed.
- The process the developer will use is summarized as follows:
  - Gain a complete understanding of the user's mental model based upon:
    - The user's needs and the user's profile.
    - A user task analysis.
  - Develop a conceptual model of the system based upon the user's mental model that includes:
    - Defining objects.
    - Developing metaphors.

2.7.1 Understanding the User's Mental Model

- Next phase in interface design is to thoroughly describe the expected system user or users and their current tasks.
- expected system users derived from the kinds of information collected in "Understand the User or Client" step and the requirements analysis techniques.
- goal of task analysis, and understanding the user - gain a picture of the user's mental model.
- mental model is an internal representation of a person's current conceptualization and understanding of something.
- Mental models are gradually developed in order to understand, explain, and do something.
- Mental models enable a person to predict the actions necessary to do things if the actions have been forgotten or have not yet been encountered.

Performing a Task Analysis

- User activities precisely described in task analysis.
- Task analysis involves breaking down the user's activities to the individual task level.
- goal is to obtain an understanding of why and how people currently do the things that will be automated.
- Knowing why establishes the major work goals; knowing how provides details of actions performed to accomplish these goals.
- Task analysis also provides information concerning workflows, the interrelationships between people, objects, and actions, and the user's conceptual frameworks.
- output of task analysis is complete description of all user tasks and interactions.
- Work activities are studied and/or described by users using the techniques just reviewed; direct observation, interviews, questionnaires, or obtaining measurements of actual current system usage.
- Measurements, for example, may be obtained for the frequency with which tasks are performed or the number of errors that are made.
- Results of task analysis:
  - listing the user's current tasks. It should be well documented and maintained. Changes in task requirements can then be easily incorporated as design iteration occurs.
  - list of objects the users see as important to what they do.
- Objects can be sorted into the following categories:
  - Concrete objects — things that can be touched.
  - People who are the object of sentences — normally organization employees, customers.

- Forms or journals — things that keep track of information.
- People who are the subject of sentences — normally the users of a system.
- Abstract objects — anything not included above.

2.7.2 Developing Conceptual Models

- output of task analysis is the creation of a conceptual model for the user interface by the designer.
- Conceptual model is the general conceptual framework through which the system's functions are presented.
- It describes how the interface will present objects, the relationships between objects, the properties of objects, and the actions that will be performed.
- A conceptual model is based on the user's mental model.
- mental model refers to a person's current level of knowledge about something and so people will *always* have them.
- mental models are influenced by a person's experiences, and people have different experiences.
- No two user mental models are likely to be exactly the same. Each person looks at the interface from a slightly different perspective.
- Goal of the designer is to facilitate for the user the development of useful *mental model of the system*.
- This designer goal accomplished by presenting to the user a *meaningful conceptual model of the system*.
- When the user then encounters the system, his or her *existing mental model* will, hopefully, mesh well with the system's conceptual model.
- As a person works with a system, he or she then develops a *mental model of the system*.
- system mental model the user derives is based upon system's behavior, including factors such as the system inputs, actions, outputs (including screens and messages), and its feedback and guidance characteristics, all of which are components of the conceptual model.
- Documentation and training also play a formative role.
- Mental models developed regardless of the particular design of a system, and will be modified with experience.
- Avoid during design creating for the user a conceptual model that leads to the creation of a false mental model of the system, or that inhibits the user from creating a meaningful or efficient mental model.

Guidelines for Designing Conceptual Models

1. **Reflect the user's mental model, not the designer's.**
   - A user will have different expectations and levels of knowledge than the designer. So, the mental models of the user and designer will be different.
   - user concerned with the task to be performed, the business objectives that must be fulfilled.
   - designer's model focused on the design of the interface, the kinds of objects, the interaction methods, and the visual representations on the screen.
   - Objects must be defined, along with their relationships, behaviors, and properties. Interaction methods must also be defined, such as input mechanisms, interaction techniques, and the contents of menus.
   - Visual screen representations must also be created, including functionality and appearance.
2. **Draw physical analogies or present metaphors.**
   - Replicate what is familiar and well known. Duplicate actions that are already well learned.
   - The success of graphical systems can be attributed, in part, to their employing the desktop metaphor.
   - A metaphor, to be effective, must be widely applicable within an interface.
   - Metaphors that are only partially or occasionally applicable should not be used.
   - In the event that a metaphor cannot be explicitly employed in a new interface, structure the new interface in terms of familiar aspects from the manual world.
3. **Comply with expectancies, habits, routines, and stereotypes.**
   - Create a system that builds on knowledge, habits, routines, and expectancies that already exist.
   - Use familiar associations, avoiding the new and unfamiliar. With color, for example, accepted meanings for red, yellow, and green are already well established.
   - Use words and symbols in their customary ways.
   - Replicate the language of the user, and create icons reflecting already known images.
4. **Provide action-response compatibility.**
   - All system responses should be compatible with the actions that elicit them.
   - Names of commands, for example, should reflect the actions that will occur.
   - organization of keys in documentation or help screens should reflect the ordering that actually exists on the keyboard.
5. **Make invisible parts of the system visible**.

- Systems composed of parts and processes, many of which are invisible to the user.
- a person must make a hypothesis about what is invisible and how it relates to what is visible.
- New users of a system make erroneous or incomplete assumptions about what is invisible and develop a faulty mental model.
- With experience gained, mental models evolve to become more accurate and complete.
- Making invisible parts of a system visible speeds up the process of developing correct mental models.
- An example of a process being made visible can be illustrated by moving a document between files.
  - ✓ In a command language interface, the document must be moved through a series of typed commands. The file is moved invisibly, and the user assumes correctly, unless an error message is received.
  - ✓ In a graphical direct manipulation system, the entire process is visible, with the user literally picking up the file in one folder by clicking on it, and dragging it to another folder.

6. Provide Proper and Correct Feedback.

- provide feedback. Keep a person informed of what is happening, and what has happened, at all times, including:
  - *Provide a continuous indication of status*. Mental models are difficult to develop if things happen, or are completed, unknown to the user. During long processing sequences, for example, interim status messages such as loading, "opening . . ." or "searching . . ." reassure the user and enable him or her to understand internal processes and more accurately predict how long something will take. Such messages also permit the pinpointing of problems if they occur.
  - *Provide visible results of actions*. For example, highlight selected objects, display new locations of moved objects, and show files that are closed.
  - *Display actions in progress*. For example, show a window that is being changed in size actually changing, not simply the window in its changed form. This will strengthen cause-and-effect relationships in the mental model.
  - *Present as much context information as possible*. To promote contextual understanding, present as much background or historical information as possible. For example, on a menu screen or in navigation, maintain a listing of the choices selected

to get to the current point. On a query or search screen, show the query or search criteria when displaying the results.

- *Provide clear, constructive, and correct error messages*. Incomplete or misleading error messages may cause false assumptions that violate and weaken the user's mental model. Error messages should always be structured to reinforce the mental model. For example, error messages addressing an incomplete action should specify *exactly* what is missing, not simply advise a person that something is incomplete.

7. **Avoid the unnecessary or irrelevant.**
   - Never display irrelevant information on the screen. People may try to interpret it and integrate it into their mental models, thereby creating a false one.
   - Irrelevant information might be unneeded data fields, screen controls, system status codes, or error message numbers.
   - If potentially misleading information cannot be avoided, point this out to the user.
   - do not overuse display techniques, or use them in meaningless ways. Too much color, for example, may distract people and cause them to make erroneous assumptions as they try to interpret the colors.
   - result will be a faulty and unclear mental model.
8. **Provide design consistency.**
   - Design consistency reduces the number of concepts to be learned.
   - Inconsistency requires the mastery of multiple models.
   - If an occasional inconsistency cannot be avoided, explain it to the user. For example, if an error is caused by a user action that is inconsistent with other similar actions, explain in the error message that this condition exists.
   - prevents user from falsely assuming that the model being operated under is incorrect.
9. **Provide documentation and a help system that will reinforce the conceptual model.**
   - Consistencies and metaphors should be explicitly described in the user documentation.
   - assists a person in learning the system.
   - Do not rely on the people to uncover consistencies and metaphors themselves.
   - help system should offer advice aimed at improving mental models.
10. **Promote the development of both novice and expert mental models.**
    - Novices and experts are likely to bring to bear different mental models when using a system.
    - easier for novices to form an initial system mental model if they are protected from the full complexity of a system.

- Employ levels of functionality that can be revealed through progressive disclosure.

Defining Objects

- Determine all objects that have to be manipulated to get work done.
  - — All *objects* that have to be manipulated to get work done must be clearly described.
  - — behavioral characteristics must be established and the attributes that differentiate each kind of object must be identified.
  - — relationship of objects to each other and the people using them must be determined.
  - — actions people take on objects must also be described.
  - — The objects to which actions apply.
  - — State information or attributes that each object in the task must preserve, display, or allow to be edited must be defined.
- Identify the objects and actions that appear most often in the workflow. go through the workflow document highlighting all nouns and verbs associated with nouns.
- Frequently appearing nouns are possible major objects. Frequently appearing verbs are actions pointing to possible major objects.
- Make the several most important objects very obvious and easy to manipulate.

Developing Metaphors

- A *metaphor* is a concept where one's body of knowledge about one thing is used to understand something else.
- Metaphors act as building blocks of a system, aiding understanding of how a system works and is organized.
- Select a metaphor or analogy for the defined objects. Choose the analogy that works best for the objects and their actions.
- Real-world metaphors most often the best choice. Replicate what is familiar and well known.
- Duplicate actions that are already well learned. If a faster or better way exists to do something, use it.
- Use simple metaphors, as they are most powerful.
- Use common metaphors; uniqueness adds complexity.
- Multiple metaphors may coexist.
- Use major metaphors even if you can't exactly replicate them visually on the screen. Exactly mimicking the real world does not always aid understanding. A representation will be satisfactory.
- Finally, test the selected metaphors.

- Do they match one's expectations and experiences? Are they easily understood or quickly learned? Change them, if testing deems it necessary.
- A common metaphor in a graphical system is the desktop and its components, items such as folders and a trash bin.
- The Web utilizes a library metaphor for the activities of browsing and searching.
- Browsing in a library occurs when you wander around book stacks looking for something interesting to read. When searching you devise an active plan to find some specific information.

2.8 THE USER'S NEW MENTAL MODEL

- When the system is implemented, and a person interacts with the new system and its interface, an attempt will be made by the person to understand the system based upon the existing mental model brought to the interaction.
- If the designer has correctly reflected the user's mental model in design, the user's mental model is reinforced and a feeling that the interface is intuitive will likely develop.
- Continued interaction with the system may influence and modify the user's concept of the system, and his or her mental model may be modified as well.
- Refinement of this mental model, a normal process, is aided by well-defined distinctions between objects and being consistent across all aspects of the interface.
- new system not accurately reflecting the user's existing mental model results in breakdowns in understanding, confusion, errors, loss of trust, and frustration.
- It can result in an inability to perform the task or job.
- When system designers have known in advance there was a gap between their conceptual model and the mental model the user would bring to the new system, designers have tried to bridge this gap through extensive documentation and training.
- The problems with this approach are: people are unproductive while being trained, people rarely read the documentation and training materials, and, even if the training material is read, the material is presented out of context.
- This creates difficulties for the users in understanding the material's relevance to their needs and goals.

2.9 DESIGN STANDARDS OR STYLE GUIDES

- A design standard or style guide documents an agreed-upon way of doing something.
- In interface design it describes the appearance and behavior of the interface and provides some guidance on the proper use of system components.

- It also defines the interface standards, rules, guidelines, and conventions that must be followed in detailed design.
- It will be based on the characteristics of the system's hardware and software, the principles of good interface and screen design, the needs of system users, and any unique company or organization requirements that may exist.

2.9.1 Value of Standards and Guidelines

- Developing and applying design standards or guidelines achieves design consistency.
- valuable to *users* because the standards and guidelines:
  - Allow faster performance.
  - Reduce errors.
  - Reduce training time.
  - Foster better system utilization.
  - Improve satisfaction.
  - Improve system acceptance.
- valuable to system *developers* because they:
  - Increase visibility of the human-computer interface.
  - Simplify design.
  - Provide more programming and design aids, reducing programming time.
  - Reduce redundant effort.
  - Reduce training time.
  - Provide a benchmark for quality control testing.

Business System Interface Standards and Guidelines

- some businesses and organizations developed and implemented human computer interface design standards.
- guideline documents were prepared for internal company or organization use only; others were published for general consumption by companies such as IBM (1987), Sun Microsystems (1990), Apple Computer (1992b), and Microsoft (1992).
- These guidelines were updated and many of these interface guidelines are published on the Web as well.
- Concurrently government and trade organizations also began working on developing interface guidelines and standards.

- Organizations addressing these issues have included the International Standards Organization (ISO), the American National Standards Institute (ANSI), and the Human Factors and Ergonomics Society.
- Standards conformance problems identified include difficulties in finding information being sought, difficulties in interpreting information, and numerous rules violations. e rules in the screen design standard were broken, found a very positive designer attitude toward the standard, both in terms of its value and content.
- Rules in the screen design standard were not adhered to for the following reasons:
  - An alternative design solution was better than that mandated by the standard.
  - Available development tools did not allow compliance with the standard.
  - Compliance with the standard was planned, but time was not yet available to implement it.
  - The rule that was broken was not known or was overlooked.

Web Guidelines and Style Guides

- Web interface design issues have also unleashed a plethora of Web-specific design guidelines and style guides, many of which are found on the Web itself.
- These guidelines can be seen on the sites of the various computer companies and interface consulting firms, in newsletters, and even on personal Web sites.
- While many of the traditional interface guidelines are applicable in a Web environment, the Web does impose a host of additional considerations.
- The haste (hurry) to publish Web design guidelines has been fueled by the explosive growth of the Web and a corresponding explosive growth in the number of developers creating sites for public access.
- There has not been an opportunity for conventions and style guides to be properly developed for Web and then accepted by the development community.
- Many Web developers have had limited knowledge of traditional interface issues and concerns, and many are unfamiliar with the traditional interface design guidelines.
- Web guideline documents have attempted to fill this void.
- Since a Web user can freely move among a seemingly endless supply of sites, no site will be seen in isolation.
- Commonality is of even greater importance than in business systems, where movement occurs between small numbers of applications.

- Today, many uniquely Web standards and guidelines are evolving by trial and error. Things are being tried to see what works best.
- De facto standards are being established when an overwhelming majority of big sites focus on one way to do something. An example is a menu bar that now frequently appears down the left side of the page.
- Standards and conventions will continue to evolve with experience and as the results of usability research become available.
- Worldwide standards are also being looked at by organizations such as the World Wide Web Consortium (2001).

2.9.2 Document Design

1. **Checklists and rationale.**
   - Provide checklists for presenting key principles and guidelines.
   - Checklists permit ease in scanning, ease in referring to key points, and make a document more readable by breaking up long sequences of text.
   - provide a rationale for why the particular guidelines should be used.
   - Understanding reasoning increases guideline acceptance. Important if the guideline is a deviation from a previous design practice.
   - when two or more design alternatives exist, provide a rationale describing the conditions under which the alternatives are appropriate.
   - It may not be easy for designers to infer when various alternatives are appropriate.

2. **Concrete examples.**
   - An effective guideline must include many concrete examples of correct design.
   - Learning by imitation is often a way we learn.

3. **Document design and access.**
   - design the document (paper or electronic) always by following recognized principles for good document design. This greatly enhances readability.
   - Provide good access mechanisms such as a thorough index, a table of contents, glossaries, and checklists.
   - An unattractive or hard to use document will not be inviting and consequently will not be used.

2.9.3 Design Support and Implementation

1. Available Reference Sources.

- Use all the available reference design sources in creating guidelines.
- References include text books on user interface design, project-specific guidelines, and the style guides for interface design and Web design created by companies such as Apple, IBM, Microsoft, and Sun.
- Other reference sources that meet your needs should also be utilized.

2. Tools.

- Use tools that support implementation of the guidelines you have established.
- Development tools make the design process much easier.
- design tools that cannot support the guideline cannot be adhered to.

3. Applying the Guidelines.

- "Is it too late to develop and implement standards?" and "What will be the impact on systems and screens now being used?"
- To address the above questions, researchers reformatted several alphanumeric inquiry screens to improve their comprehensibility and readability.
- For expert system users, the reformatted screens reduced errors but decision making time remained the same.
- For novice system users, the reformatted screens improved learning speed and accuracy.
- it appears, that changes that enhance screens will benefit *both* novice and expert users already familiar with the current screens.
- It is never too late to begin to change.

QUESTION BANK:

1. List and explain the pitfalls in the development path of design process. **(8 Marks)**
2. Explain few significant direct techniques for determining business requirements. **(8 Marks)**
3. Write a note on guidelines that must be followed during detailed interface design that are valuable for users and developers. **(8 Marks)**
4. List and explain the five commandments for designing for people. **(6 Marks)**
5. Describe in details the important human characteristics in user interface design (any five). **(10 Marks)**
6. Explain the common usability problems in web based systems. **(6 Marks)**
7. Explain the techniques for determining user requirements using indirect methods. **(10 Marks)**
8. Briefly explain five commandments to eliminate the pitfalls in designing the interface. (**8 Marks)**
9. Explain the objective criteria for measuring usability. **(8 Marks)**
10. Explain briefly about human interaction speeds. **(8 Marks)**
11. Explain the guidelines for designing conceptual models. **(8 Marks)**
12. Write a note on guidelines that must be followed during interface design that are valuable for users. **(8 Marks)**
13. Write short notes on Human Interaction speeds. **(6 Marks)**
14. Discuss with suitable examples the human characteristics on design. **(10 Marks)**
15. Discuss the models for determining basic business functions. **(10 Marks)**