UNIT I

INTRODUCTION, THE GRAPHICAL USER INTERFACE, WEB USER

Introduction: Importance of user Interface: Definition, Importance of Good Design, Benefits of Good Design, A Brief History of Screen Design.

The Graphical User Interface: Popularity of Graphics, the Concept of Direct Manipulation, Graphical System, Characteristics.

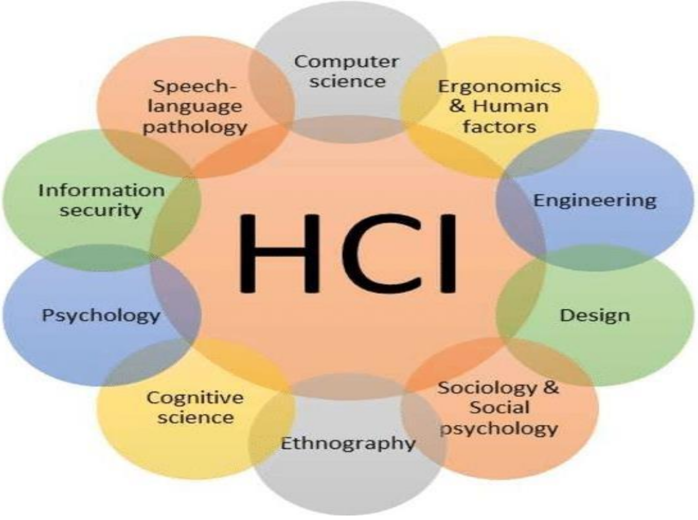
Web User – Interface Popularity, Characteristics- Principles of User Interface.

INTRODUCTION:

DEFINITION:

- "**Human-computer interaction** is a discipline concerned with the **design, evaluation and implementation** of interactive computing systems for human use and with the study of major phenomena surrounding them".
- Human–computer interaction (HCI), alternatively **man–machine interaction (MMI)** or **computer–human interaction (CHI)** is the study of **interaction between people (users) and computers**.

HCI is an interdisciplinary field:



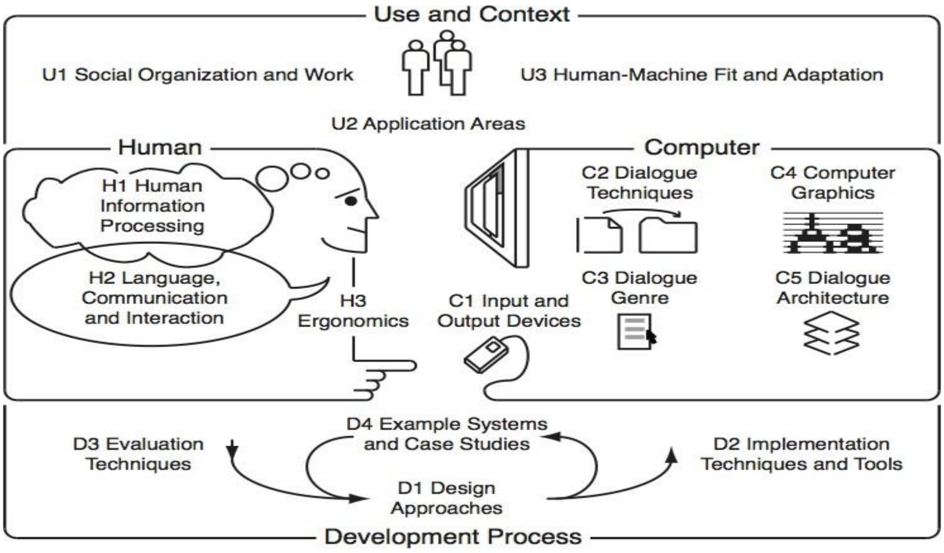
GOALS of HCI:

- A basic goal of HCI is
  – to improve the interactions between users and computers
  – by making computers more usable and receptive to the user's needs.
- A long term goal of HCI is
  – to design systems that minimize the barrier between the human's cognitive model of what they want.
  – to accomplish and the computer's understanding of the user's task.

Why HCI?

- In the past, computers were expensive & used by technical people only.
- Now, computers are cheap and used by non technical people (different backgrounds, needs, knowledge, Skills).
- Computer and software manufacturers have noticed the importance of making computers user friendly: easy to use, save people time etc.

HCI Scope:


Use and Context
U1 Social Organization and Work
U3 Human-Machine Fit and Adaptation
U2 Application Areas
Human
H1 Human Information Processing
H2 Language, Communication and Interaction
H3 Ergonomics
Computer
C2 Dialogue Techniques
C4 Computer Graphics
C3 Dialogue Genre
C5 Dialogue Architecture
C1 Input and Output Devices
D3 Evaluation Techniques
D4 Example Systems and Case Studies
D2 Implementation Techniques and Tools
D1 Design Approaches
Development Process

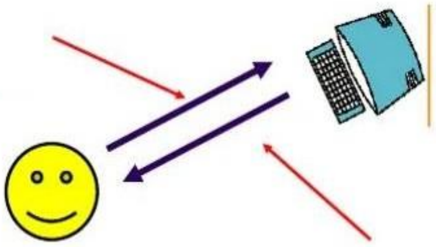
HCI Scope contents:

Use & Context: Find application areas of computers.

- **Human:** Study psychological & physiological aspects, e.g., study how a user learns to use a new product, study human typing speed.
- **Computer:** Hardware & Software offered, e.g., input & output devices, speed, interaction styles, computer graphics.
- **Development:** Design, Implementation & evaluation.

Components of HCI:

- The goal of HCI is to improve the interaction between users and computers by making computers more user-friendly and receptive to the user’s needs.
  - Human
  - Computer Interaction



- **Human:**
  - Individual user, a group of users working together, a sequence of users in an organization.
- **Computer:**
  - Desktop computer, large-scale computer system, Pocket PC, embedded system (e.g., photocopier, microwave oven) Software (e.g., search engine, word processor).
- **User Interface:**
  - Parts of the computer that the user contacts with.
- **Interaction:**
  - Usually involve a dialog with feedback & control throughout performing a task (e.g., user invokes print command and then interface replies with a dialog box).

DEFINING THE USER INTERFACE:

- **User interface design** is a subset of a field of study called ***human-computer interaction*** **(HCI)**.
- User Interfaces are the access points where users interact with designs. They come in **three** formats:
  - Graphical user interfaces (GUIs)
  - Voice controlled interfaces (VUIs)
  - Gesture based interfaces
- HCI designers must consider a variety of factors:
  - What people want and expect,
  - What physical limitations and abilities people possess,
  - How their perceptual and information processing systems work, and
  - what people find enjoyable and attractive.
- Designers must also consider technical characteristics and limitations of the computer hardware and software.
- The *user interface* is the part of a computer and its software that people can **see, hear, touch,talk to, or otherwise understand or direct.**
- The user interface has essentially two components: **input and output**.
  - ***Input*** is how a person communicates his or her needs or desires to the computer. Some common input components are the keyboard, mouse, trackball, one's finger (for touch-sensitive screensor pads), and one's voice (for spoken instructions).
  - ***Output*** is how the computer conveys the results of its computations and requirements to theuser.
- Proper interface design will provide a **mix of well-designed input and output mechanisms** that satisfy the user's needs, capabilities, and limitations in the most effective way possible.

THE IMPORTANCE OF GOOD DESIGN:

- With today's technology and tools, and our motivation to create really effective and usable interfaces and screens, why do we continue to produce systems that are inefficient and confusing or, at worst, just plain unusable? Is it because:
  - We don't care?
  - We don't possess common sense?
  - We don't have the time?
  - We still don't know what really makes good design?
- A well-designed interface and screen is important to our users. It is their window to view the capabilities of the system.
- It is also the vehicle through which many critical tasks are presented.

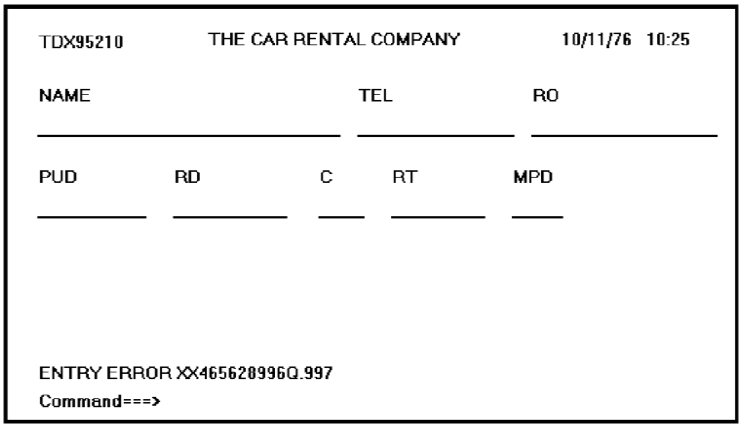
- A screen's layout and appearance affect a person in a variety of ways. If they are confusing and inefficient, people will have greater difficulty in doing their jobs and will make more mistakes.
- Poor design may even chase some people away from a system permanently. It can also lead to aggravation, frustration, and increased stress.

THE BENEFITS OF GOOD DESIGN:

- Good design
  - Screen Clarity
  - Readability
  - Screens less crowded
  - 20% more productive
- Proper formatting of information
  - Positive effect on performance
- Training costs, time are reduced.
- Support line costs are reduced.
- Employee satisfaction increased, frustration decreased.
- Ultimately customers benefit.
- Significant economic benefits and development**.**

A BRIEF HISTORY OF SCREEN DESIGN:

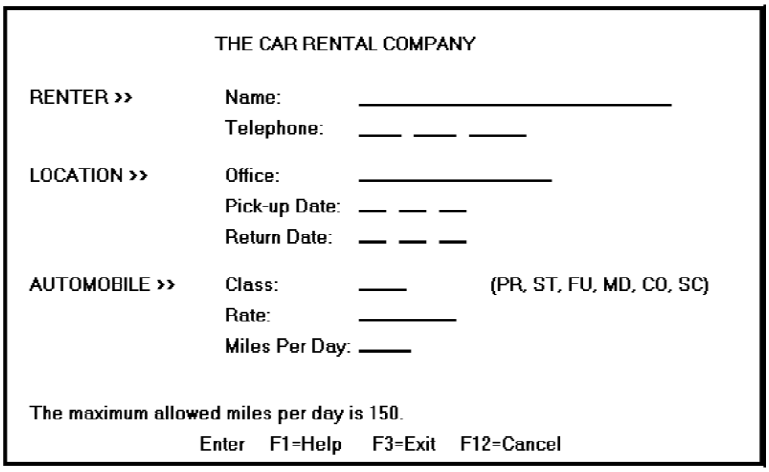
- While developers have been designing screens since a cathode ray tube display was first attached to a computer, more widespread interest in the application of good design principles to screens did not begin to emerge until the early 1970s, when IBM introduced its 3270 cathode ray tube text-based terminal.
- The 3270 was used in extremely number of ways in the office, and company-specific guidelines for good screen design occasionally began to surface (e.g., Galitz and DiMatteo, 1974).
- A1970s screen often resembled the one shown in Figure.



```
TDX95210          THE CAR RENTAL COMPANY          10/11/76   10:25

NAME                                   TEL               RO
_____________________________________  ________________  ____________________

PUD          RD               C      RT           MPD
___________  ____________     ____   __________   _____



ENTRY ERROR XX465628996Q.997
Command===>
```

- It usually consisted of many fields with very cryptic and often unintelligible captions. It was visually cluttered and often possessed a command field that challenged the user to remember what had to be keyed into it.

- Effectively using this kind of screen required a great deal of practice and patience. Most early screens were monochromatic, typically presenting green text on black backgrounds.
- At the turn of the decade, guidelines for text-based screen design were finally made widely available (Galitz, 1980, 1981) and many screens began to take on a much less cluttered look through concepts such as grouping and alignment of elements, as shown in Figure.



THE CAR RENTAL COMPANY

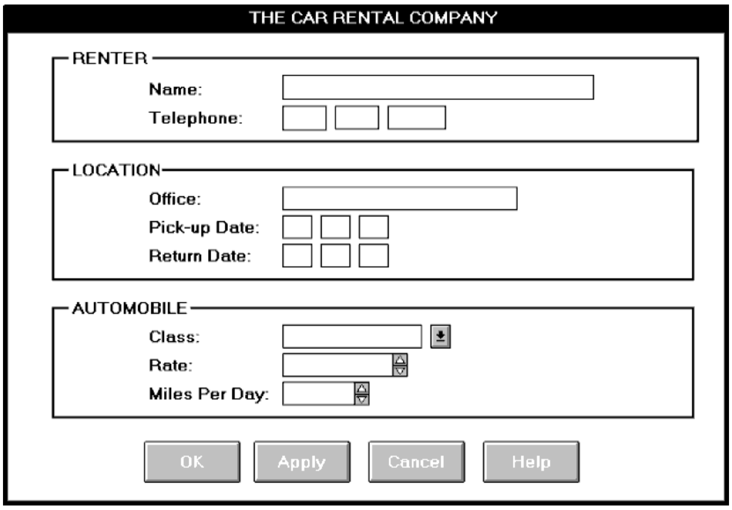
RENTER >> Name: ____________
Telephone: ___ ___ ____

LOCATION >> Office: ________
Pick-up Date: __ __ __
Return Date: __ __ __

AUTOMOBILE >> Class: ___ (PR, ST, FU, MD, CO, SC)
Rate: _____
Miles Per Day: ___

The maximum allowed miles per day is 150.
Enter F1=Help F3=Exit F12=Cancel

- User memory was supported by providing clear and meaningful field captions and by listing commands on the screen, and enabling them to be applied through function keys. Messages also became clearer. These screens were not entirely clutter-free, however. Instructions and reminders to the user had to be inscribed on the screen in the form of prompts or completion aids such as the codes PR and SC.
- The advent of graphics yielded another milestone in the evolution of screen design, as shown in Figure.



THE CAR RENTAL COMPANY

RENTER
Name:
Telephone:

LOCATION
Office:
Pick-up Date:
Return Date:

AUTOMOBILE
Class:
Rate:
Miles Per Day:

- While some basic design principles did not change, such as groupings and alignment, borders were made available to visually enhance groupings, and buttons and menus for implementing commands replaced function keys.

- Multiple properties of elements were also provided, including different font sizes and styles, line thickness, and colors. The entry field was supplemented by many other kinds of controls, including list boxes, drop-down combination boxes, spin boxes, and so forth.
- These new controls were much more effective in supporting a person's memory, now simply allowing for selection from a list instead of requiring a remembered key entry. Screens could also be simplified, the much more powerful computers being able to quickly present a new screen.

THE GRAPHICAL USER INTERFACE:

- A ***user interface***, is a collection of techniques and mechanisms to interact with something.
- In a *graphical* interface, the primary interaction mechanism is a pointing device. This device is the electronic equivalent to the human hand.
- What the user interacts with is a collection of elements referred to as ***objects***.
- People perform operations, called **actions**, on objects. The operations include **accessing** and **modifying objects by pointing, selecting, and manipulating**.

POPULARITY OF GRAPHICS:

- Older text - based screen possessed a one dimensional.
- Graphic screens assumed a three - dimensional look.
- Controls appeared to rise above the screen and move when activated.
- Information could appear, and disappear, as needed.
- Text could be replaced by graphical images called icons.
- These icons could represent objects or actions.
- Selection fields such as radio buttons, check boxes, list boxes, and palettes coexisted with the reliable old text entry field.
- More sophisticated text entry fields with attached or dropdown menus of.
- Objects and actions were selected through use of pointing mechanisms.
- Increased computer power.
- User's actions to be reacted quickly, dynamically, and meaningfully
- **WIMP interface:** windows, icons, menus, and pointers.
- Graphic presentation is much more effective than other presentation methods.
- It permits faster information transfer between computers and people by permitting more visual comparisons of amounts, trends, or relationships; more compact representation of information.

THE CONCEPT OF DIRECT MANIPULATION:

- The term used to describe graphical systems with this style of interaction was first used by **Shneiderman (1982)**. He called them "**direct manipulation**" systems, suggesting that they possess the following characteristics:

The system is portrayed as an extension of the real world.

- It is assumed that a person is already familiar with the objects and actions in his or her environment of interest. The system simply replicates them and portrays them on a different medium, the screen.
- A person has the power to access and modify these objects, including windows.
- A person is allowed to work in a familiar environment and in a familiar way, focusing on the data, not the application and tools. The physical organization of the system, which most often is unfamiliar, is hidden from view and is not a distraction.

Objects and actions are continuously visible.

- Reminders of actions to be performed are also obvious, where labeled buttons replace complex syntax and command names.
- Cursor action and motion occurs in physically obvious and intuitively natural ways.
- **Nelson (1980)** described this concept as ***virtual reality***, a representation of reality that can be manipulated.
- **Hatfield (1981)** is credited with calling it **WYSIWYG** (what you see is what you get).
- **Rutkowski (1982)** described it as *transparency*, where one's intellect is applied to the task,not the tool.

Actions are rapid and incremental with visible display of results.

- Because tactile feedback is not yet possible, the results of actions are immediately displayed visually on the screen in their new and current form.
- Auditory feedback may also be provided.
- The impact of a previous action is quickly seen, and the evolution of tasks is continuous and effortless.

Incremental actions are easily reversible.

- Finally, actions, if discovered to be incorrect or notdesired, can be easily undone.

Earlier Direct Manipulation Systems

- The concept of direct manipulation actually preceded the first graphical system. The earliest full-screen text editors possessed similar characteristics.
- Screens of text resembling a piece of paper on one's desk could be created (extension of real world) and then reviewed in their entirety (continuous visibility).

- Editing or restructuring could be easily accomplished (through rapid incremental actions) and the results immediately seen. Actions could be reversed when necessary. It took the advent of graphical systems to crystallize the direct manipulation concept, however.

Indirect Manipulation

- In practice, direct manipulation of *all* screen objects and actions may not be feasible because of the following:
    - The operation may be difficult to conceptualize in the graphical system.
    - The graphics capability of the system may be limited.
    - The amount of space available for placing manipulation controls in the window border may be limited.
    - It may be difficult for people to learn and remember all of the necessary operations and actions.
- When this occurs, *indirect manipulation* is provided. Indirect manipulation substitutes words and text, such as pull-down or pop-up menus, for symbols, and substitutes typing for pointing. Most window systems are a combination of both direct and indirect manipulation.
- A menu may be accessed by pointing at a menu icon and then selecting it (direct manipulation).
- The menu itself, however, is a textual list of operations (indirect manipulation).
- When an operation is selected from the list, by pointing or typing, the system executes it as a command.

GRAPHICAL SYSTEMS:

Advantages and Disadvantages:

Advantages

- Symbols recognized faster than text
- Faster learning
- Faster use and problem solving
- Easier remembering
- More natural
- Exploits visual/spatial cues.
- Fosters more concrete thinking
- Provides context
- Fewer errors
- Immediate feedback
- Predictable system responses
- Easily reversible actions.
- Less anxiety concerning use.
- More attractive

- May consume less space
- Easily augmented with text displays
- Smooth transition from command language system

Disadvantages

- Greater design complexity.
- Learning still necessary
- Lack of experimentally-derived design guidelines
- Use a pointing device may also have to be learned
- Human comprehension limitations
- Window manipulation requirements
- Production limitations
- Few tested icons exist
- Inefficient for touch typists
- Inefficient for expert users
- Not always the preferred style of interaction
- Not always fastest style of interaction
- Increased chances of clutter and confusion
- May consume more screen space
- Hardware limitations

CHARACTERISTICS OF THE GRAPHICAL USER INTERFACE:

- A graphical system possesses a set of defining concepts that includes sophisticated visual presentation, pick-and-click interaction, a restricted set of interface options, visualization, object orientation, use of a person’s recognition memory, and concurrent performance.

Sophisticated Visual Presentation

- Visual presentation is the visual aspect of the interface. It is what people see on the screen. The sophistication of a graphical system permits displaying lines, including drawings and icons.
- It also permits the displaying of a variety of character fonts, including different sizes and styles.
- The display of 16 million or more colors is possible on some screens. Graphics also permit animation and the presentation of photographs and motion video.
- The meaningful interface elements visually presented to the user in a graphical system include windows, menus, icons to represent objects such as programs or files, assorted screen-based controls, a mouse or other pointing device, and the cursor.

- The objective is to reflect visually on the screen the real world of the user as realistically, meaningfully, simply, and clearly as possible.

Pick-and-Click Interaction

- Elements of a graphical screen upon which some action is to be performed must first be identified.
- The motor activity required of a person to identify this element for a proposed action is commonly referred to as *pick*, and the signal to perform an action as *click*.
- The primary mechanism for performing this pick-and-click is most often the mouse and its buttons. The user moves the mouse pointer to the relevant element (pick) and the action is signaled (click).
- The secondary mechanism for performing these selection actions is the keyboard. Most systemspermit pick-and-click to be performed using the keyboard as well.

Restricted Set of Interface Options

- The array of alternatives available to the user is what is presented on the screen or what may be retrieved through what is presented on the screen — nothing less, nothing more.
- This concept fostered the acronym WYSIWYG.

Visualization

- Visualization is a cognitive process that enables people to understand information that is difficult to perceive, because it is either too voluminous or too abstract.
- Presenting specialized graphic facilitates visualization. The best visualization method for an activity depends on what people are trying to learn from the data.
- Effective visualizations can facilitate mental insights, increase productivity, and more accurate use of data.

Object Orientation

- A graphical system consists of objects and actions. *Objects* are what people see on the screen. They are manipulated as a single unit. Objects can be composed of *subobjects*.
- IBM’s System Application Architecture Common User Access (SAA CUA) Advanced Interface Design Reference (SAA CUA) (IBM, 1991) breaks objects into three meaningful classes: **data, container, and device.**
- **Data objects** present information. This information, either text or graphics, normally appears in the body of the screen.
- **Container objects** are objects that hold other objects. They are used to group two or more related objects for easy access and retrieval.
- There are three kinds of **container objects**: **the workplace, folders, and work area**s. The **workplace** is the desktop, the storage area for all objects.

- **Folders** are general-purpose containers for long-term storage of objects. **Work areas** are temporary storage folders used for storing multiple objects currently being worked on.
- **Device objects** represent physical objects in the real world, such as printers. These objects may contain others for acting upon. A file, may be placed in a printer for printing of its contents.
- Microsoft Windows specifies the characteristics of objects depending upon the relationships that exist between them.
- Objects can exist within the context of other objects, and one object may affect the way another object appears or behaves. These relationships are called **collections, constraints, composites, and containers.**
- A **collection** is the simplest relationship — the objects sharing a common aspect. A collection might be the result of a query or a multiple selection of objects.
- A **constraint** is a stronger object relationship. Changing an object in a set affects some other object in the set. A document being organized into pages is an example of a constraint.
- A **composite** exists when the relationship between objects becomes so significant that the aggregation itself can be identified as an object. Examples include a range of cells organized into a spreadsheet, or a collection of words organized into a paragraph.
- A **container** is an object in which other objects exist. Examples include text in a document or documents in a folder.
- Another important object characteristic is *persistence*. Persistence is the maintenance of a state once it is established.

Properties or Attributes of Objects

- Objects also have properties or attributes. Properties are the unique characteristics of an object. Properties help to describe an object and can be changed by users.
- Examples of properties are text styles (such as normal or italics), font sizes (such as 10 or 12 points), or window background colors (such as black or blue).

Actions

- In addition to objects, there are actions. People take actions on objects. They manipulate objectsin specific ways (commands) or modify the properties of objects (property or attribute specification).
- *Commands* are actions that manipulate objects. They may be performed in a variety of ways, including direct manipulation or a command button.
- *Property/attribute specification* actions establish or modify the attributes or properties of objects. The following is a typical property/attribute specification sequence:

1. The user selects an object such as several words of text.
2. The user then selects an action to apply to that object, such as the action Bold.
3. The selected words are made bold and will remain bold until selected and changed again.

Application versus Object or Data Orientation

- An application-oriented approach takes an action:object approach, like the following:
  Action> 1. The user opens an application such as word processing.
  Object> 2. The user then selects a file or other object such as a memo.
- An object-oriented object:action approach does the following:
  Object> 1. The user chooses an object such as a memo.
  Action> 2. The user then selects an application such as word processing.

Views

- Views are ways of looking at an object's information. IBM's SAA CUA describes four kinds of views: **composed, contents, settings, and help**.
  - **Composed** views present information and the objects contained within an object. They are typically associated with data objects and are specific to tasks and products being worked with.
  - **Contents** views list the components of objects.
  - **Settings** views permit seeing and changing object properties.
  - **Help** views provide all of the help functions.

Use of Recognition Memory

- Continuous visibility of objects and actions encourages use of a person's more powerful recognition memory. This eliminates the "**out of sight, out of mind**" problem.

Concurrent Performance of Functions

- Graphic systems may do two or more things at one time. Multiple programs may run simultaneously. When a system is not busy on a primary task, it may process background tasks (cooperative multitasking).
- Data may also be transferred between programs. It may be temporarily stored on a clipboard for later transfer or automatically swapped between programs.

The Web User Interface

- The expansion of the World Wide Web since the early 1990s.
- Once simply a communication medium for **scientists and researchers**, have spread deeply into **businesses, organizations, and homes** around the world.

- Web interface design is essentially the **design of navigation and the presentation of information.** It is about content, not data.
- Proper interface design is largely a matter of **properly balancing the structure and relationships of menus, content, and other linked documents or graphics.**
- The design goal is to build a **hierarchy of menus and pages that feels natural, is well structured, is easy to use**, and is **truthful**.
- The Web is a **navigation environment** where people move between pages of information, not an application environment.
- It is also a **graphically rich** environment.

THE POPULARITY OF THE WEB:

- While the introduction of the graphical user interface revolutionized the user interface, the Web revolutionized computing.
- It enables millions of people scattered across the globe to communicate, access information, publish, and be heard.
- It enables people to control much of the display and the rendering of Web pages. People can also change aspects such as typography and colors, turn off graphics, decide whether or not to transmit certain data over non-secure channels, and accept or refuse cookies.
- Nowhere in the history of computing has the user been given so much control. Web usage has reflected this popularity.
- The number of Internet hosts has risen dramatically. In 1984 hosts online exceeded 1,000; in 1987, 10,000; in 1989, 100,000; in 1990, 300,000; in 1992 hosts exceeded one million.
- Commercialization of the Internet saw even greater expansion of the growth rate. In 1993, Internet traffic was expanding at a 341,634 percent annual growth rate.
- In 1996, there were nearly 10 million hosts online and 40 million connected people (PBS Timeline). In 2005 the number of Internet hosts exceeded 350 million (Zakon.org, 2006), the number of users one billion (Nielsen, 2005g).
- The largest percentage of Internet users are in the Asia/Pacific region (36%) according to Morgan Stanley (2005).
- Percentage of users in other world regions are Europe (24%), North America (23%), South America (5%), and the rest of the world (12%).
- User control has had some decided disadvantages for some Web site owners as well. Users have become much more discerning about good design.

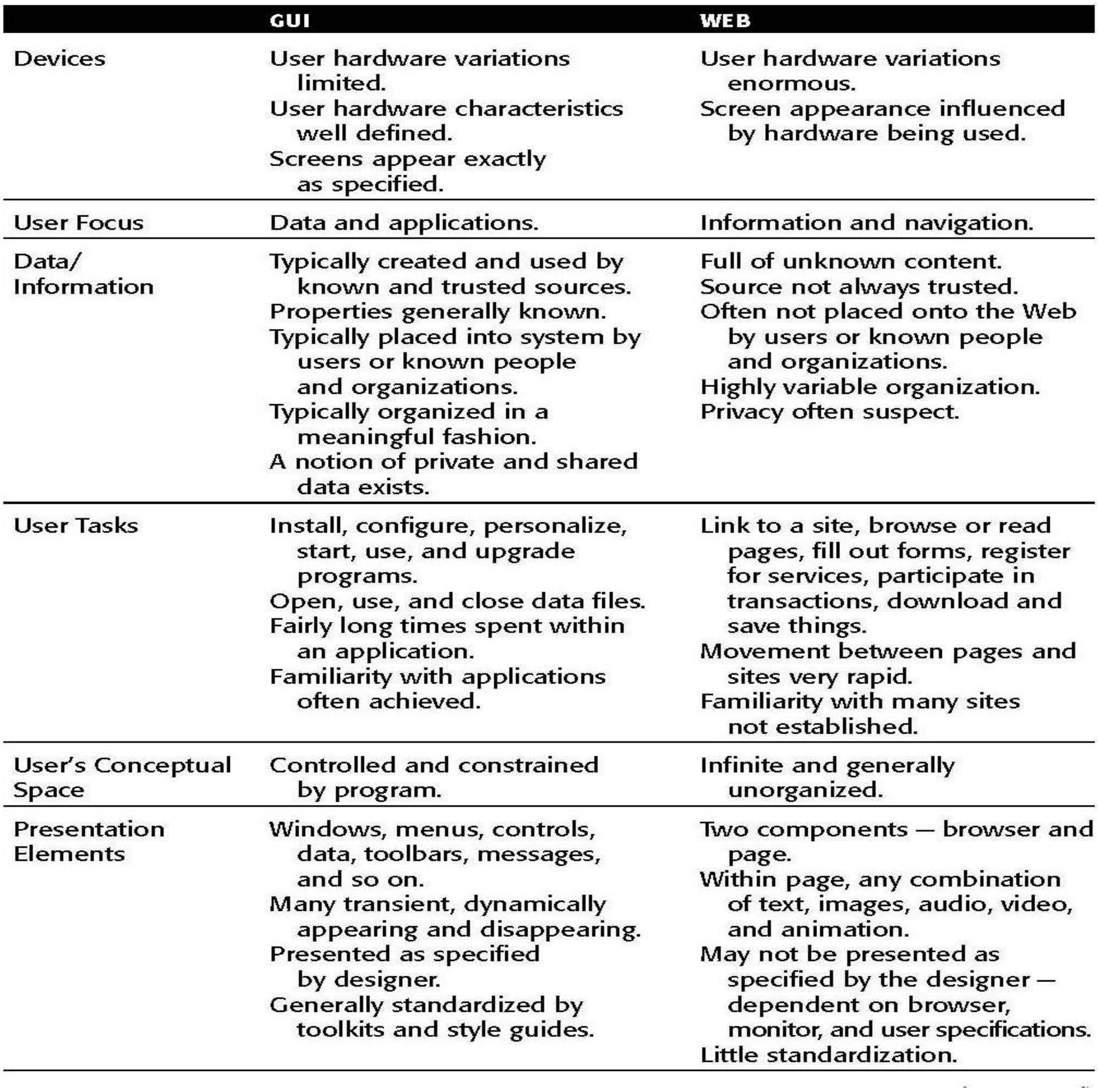
- Slow download times, confusing navigation, confusing page organization, disturbing animation, or other undesirable site features often resultin user abandonment of the site for others with a more agreeable interface.
- People are quick to vote with their mouse, and these warnings should not go unheeded.

CHARACTERISTICS OF A WEB INTERFACE:

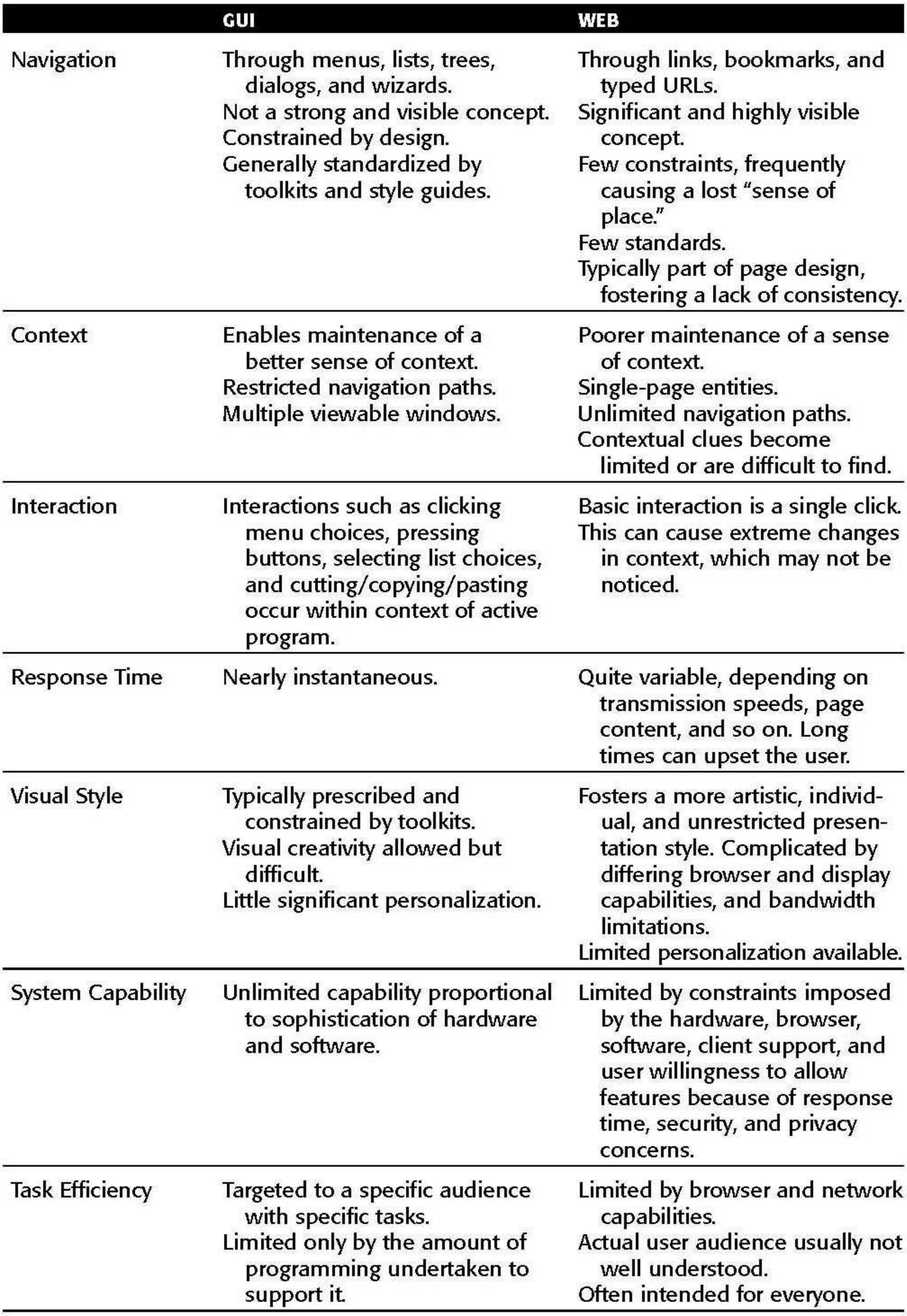
- A Web interface possesses many characteristics, some similar to a GUI interface.

GUI versus Web Page Design

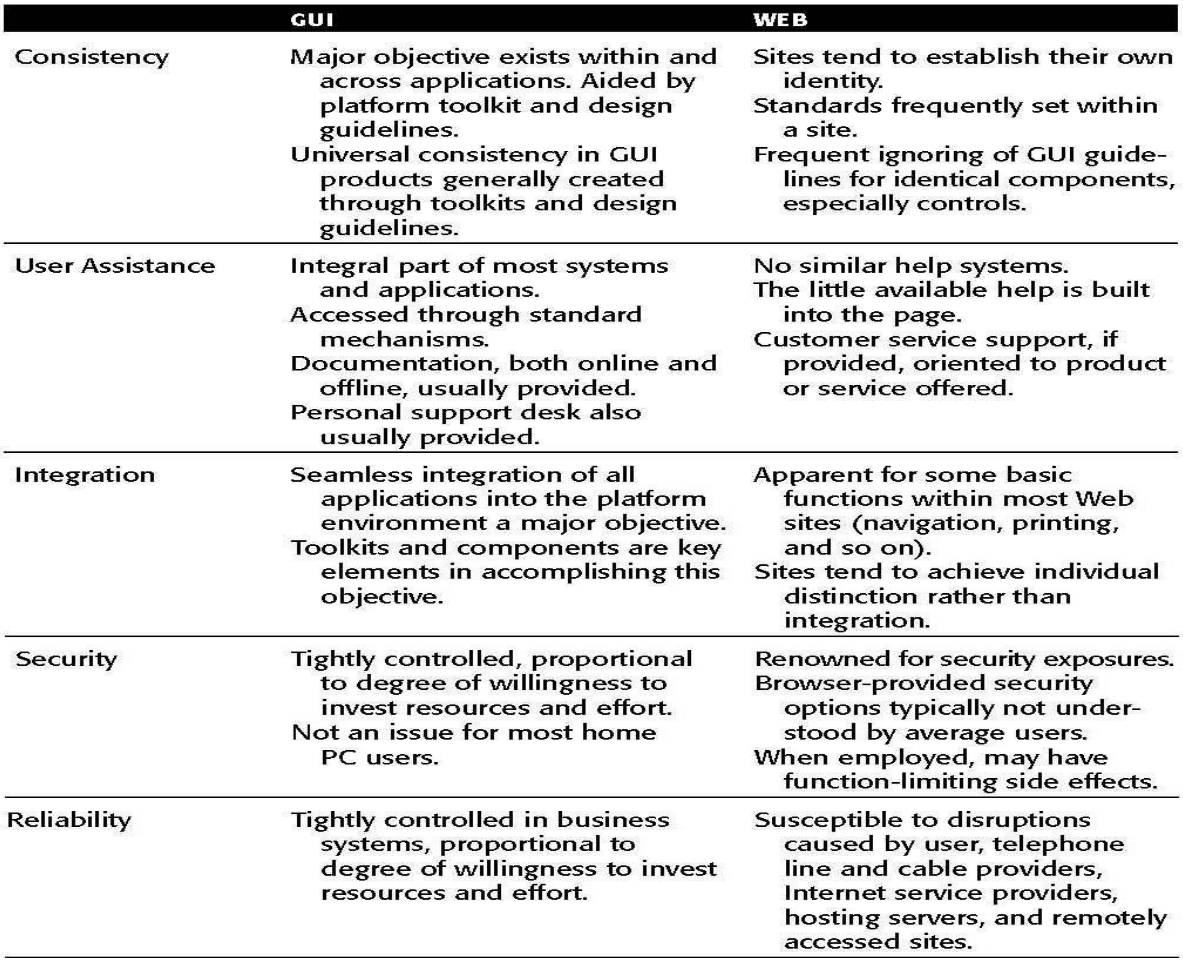
- GUI and Web interface design are similar. Both are software designs, they are used by people, they are interactive, they are heavily visual experiences presented through screens, and they are composed of many similar components.



	GUI	WEB
Devices	User hardware variations limited. User hardware characteristics well defined. Screens appear exactly as specified.	User hardware variations enormous. Screen appearance influenced by hardware being used.
User Focus	Data and applications.	Information and navigation.
Data/ Information	Typically created and used by known and trusted sources. Properties generally known. Typically placed into system by users or known people and organizations. Typically organized in a meaningful fashion. A notion of private and shared data exists.	Full of unknown content. Source not always trusted. Often not placed onto the Web by users or known people and organizations. Highly variable organization. Privacy often suspect.
User Tasks	Install, configure, personalize, start, use, and upgrade programs. Open, use, and close data files. Fairly long times spent within an application. Familiarity with applications often achieved.	Link to a site, browse or read pages, fill out forms, register for services, participate in transactions, download and save things. Movement between pages and sites very rapid. Familiarity with many sites not established.
User's Conceptual Space	Controlled and constrained by program.	Infinite and generally unorganized.
Presentation Elements	Windows, menus, controls, data, toolbars, messages, and so on. Many transient, dynamically appearing and disappearing. Presented as specified by designer. Generally standardized by toolkits and style guides.	Two components — browser and page. Within page, any combination of text, images, audio, video, and animation. May not be presented as specified by the designer — dependent on browser, monitor, and user specifications. Little standardization.



	GUI	WEB
Navigation	Through menus, lists, trees, dialogs, and wizards. Not a strong and visible concept. Constrained by design. Generally standardized by toolkits and style guides.	Through links, bookmarks, and typed URLs. Significant and highly visible concept. Few constraints, frequently causing a lost "sense of place." Few standards. Typically part of page design, fostering a lack of consistency.
Context	Enables maintenance of a better sense of context. Restricted navigation paths. Multiple viewable windows.	Poorer maintenance of a sense of context. Single-page entities. Unlimited navigation paths. Contextual clues become limited or are difficult to find.
Interaction	Interactions such as clicking menu choices, pressing buttons, selecting list choices, and cutting/copying/pasting occur within context of active program.	Basic interaction is a single click. This can cause extreme changes in context, which may not be noticed.
Response Time	Nearly instantaneous.	Quite variable, depending on transmission speeds, page content, and so on. Long times can upset the user.
Visual Style	Typically prescribed and constrained by toolkits. Visual creativity allowed but difficult. Little significant personalization.	Fosters a more artistic, individual, and unrestricted presentation style. Complicated by differing browser and display capabilities, and bandwidth limitations. Limited personalization available.
System Capability	Unlimited capability proportional to sophistication of hardware and software.	Limited by constraints imposed by the hardware, browser, software, client support, and user willingness to allow features because of response time, security, and privacy concerns.
Task Efficiency	Targeted to a specific audience with specific tasks. Limited only by the amount of programming undertaken to support it.	Limited by browser and network capabilities. Actual user audience usually not well understood. Often intended for everyone.



	GUI	WEB
Consistency	Major objective exists within and across applications. Aided by platform toolkit and design guidelines. Universal consistency in GUI products generally created through toolkits and design guidelines.	Sites tend to establish their own identity. Standards frequently set within a site. Frequent ignoring of GUI guidelines for identical components, especially controls.
User Assistance	Integral part of most systems and applications. Accessed through standard mechanisms. Documentation, both online and offline, usually provided. Personal support desk also usually provided.	No similar help systems. The little available help is built into the page. Customer service support, if provided, oriented to product or service offered.
Integration	Seamless integration of all applications into the platform environment a major objective. Toolkits and components are key elements in accomplishing this objective.	Apparent for some basic functions within most Web sites (navigation, printing, and so on). Sites tend to achieve individual distinction rather than integration.
Security	Tightly controlled, proportional to degree of willingness to invest resources and effort. Not an issue for most home PC users.	Renowned for security exposures. Browser-provided security options typically not understood by average users. When employed, may have function-limiting side effects.
Reliability	Tightly controlled in business systems, proportional to degree of willingness to invest resources and effort.	Susceptible to disruptions caused by user, telephone line and cable providers, Internet service providers, hosting servers, and remotely accessed sites.

PRINCIPLES OF USER INTERFACE DESIGN:

- An interface must really be just an extension of a person. This means that the system and its software must reflect a person's capabilities and respond to his or her specific needs.
- The interface should serve as both a **connector and a separator**: a connector in that it ties the user to the power of the computer, and a separator in that it minimizes the possibility of the participants damaging one another.
- Many principles are based on research, others on the collective thinking of behaviorists working with user interfaces.

Principles for the Xerox STAR

- The design of the Xerox STAR was guided by a set of principles that evolved over its lengthy development process (Smith, et al., 1982; Verplank, 1988). These principles established the foundation for graphical interfaces and are as follows:
- **The illusion of manipulable objects**.
    - Displayed objects that are selectable and manipulable must be created. A design challenge is to invent a set of displayable objects that are represented meaningfully and appropriately for the intended application.

  - Verplank called this “graphics with handles on it.”
- **Visual order and viewer focus**.
  - Attention must be drawn, at the proper time, to the important and relevant elements of the display.
  - Effective visual contrast between various components of the screen is used to achieve this goal
- **Revealed structure**.
  - The distance between one’s intention and the effect must be minimized.
- **Consistency**.
  - Consistency aids learning. Consistency is provided in such areas as element location; grammar; font shapes, styles, and sizes; selection indicators; and contrast and emphasis techniques.
- **Appropriate effect or emotional impact**.
  - The interface must provide the appropriate emotional effect for the product and its market.
- **A match with the medium**.
  - The interface must also reflect the capabilities of the device on which it will be displayed. Quality of screen images will be greatly affected by a device’s resolution and color-generation capabilities.

General Principles

Aesthetically Pleasing:

- Provide visual appeal by following these presentation and graphic design principles:
  - Provide meaningful contrast between screen elements.
  - Create groupings.
  - Align screen elements and groups.
  - Provide three-dimensional representation.
  - Use color and graphics effectively and simply.

Clarity:

- The interface should be visually, conceptually, and linguistically clear, including:
  - Visual elements
  - Functions
  - Metaphors
  - Words and text

Compatibility:

- Provide compatibility with the following:
  - The user

  - The task and job
  - The product
  - Adopt the user's perspective.
  - "Know the user" is the fundamental principle in interface design.

Comprehensibility:

- A system should be easily learned and understood. A user should know the following:
  - What to look at
  - What to do
  - When to do it
  - Where to do it
  - Why to do it
  - How to do it
- The flow of actions, responses, visual presentations, and information should be in a sensible order that is easy to recollect and place in context.

Configurability

- Permit easy personalization, configuration, and reconfiguration of settings.
  - Enhances a sense of control.
  - Encourages an active role in understanding.

Consistency

- A system should look, act, and operate the same throughout. Similar components should:
  - Have a similar look.
  - Have similar uses.
  - Operate similarly.
- The same action should always yield the same result.
- The function of elements should not change.
- The position of standard elements should not change.

Control

- The user must control the interaction.
  - Actions should result from explicit user requests.
  - Actions should be performed quickly.
  - Actions should be capable of interruption or termination.
  - The user should never be interrupted for errors.
- The context maintained must be from the perspective of the user.
- The means to achieve goals should be flexible and compatible with the user's skills, experiences, habits, and preferences.
- Avoid modes since they constrain the actions available to the user.
- Permit the user to customize aspects of the interface, while always providing a proper set of defaults.

Directness:

- Provide direct ways to accomplish tasks.
    - Available alternatives should be visible.
    - The effect of actions on objects should be visible.

Efficiency:

- Minimize eye and hand movements, and other control actions.
    - Transitions between various system controls should flow easily and freely.
    - Navigation paths should be as short as possible.
    - Eye movement through a screen should be obvious and sequential.
- Anticipate the user's wants and needs whenever possible.

Familiarity:

- Employ familiar concepts and use a language that is familiar to the user.
- Keep the interface natural, mimicking the user's behavior patterns.
- Use real-world metaphors.

Flexibility:

- A system must be sensitive to the differing needs of its users, enabling a level and type of performance based upon:
    - Each user's knowledge and skills.
    - Each user's experience.
    - Each user's personal preference.
    - Each user's habits.
    - The conditions at that moment.

Forgiveness:

- Tolerate and forgive common and unavoidable human errors.
- Prevent errors from occurring whenever possible.
- Protect against possible catastrophic errors.
- When an error does occur, provide constructive messages.

Predictability:

- The user should be able to anticipate the natural progression of each task.
    - Provide distinct and recognizable screen elements.
    - Provide cues to the result of an action to be performed.
- All expectations should be fulfilled uniformly and completely.

Recovery:

- A system should permit:
    - Commands or actions to be abolished or reversed.
    - Immediate return to a certain point if difficulties arise.
- Ensure that users never lose their work as a result of:
    - An error on their part.
    - Hardware, software, or communication problems.

Responsiveness:

- The system must rapidly respond to the user's requests.
- Provide immediate acknowledgment for all user actions:
    - Visual.
    - Textual.
    - Auditory.

Simplicity:

- Provide as simple an interface as possible.
- Five ways to provide simplicity:
    - Use progressive disclosure, hiding things until they are needed.
- Present common and necessary functions first.
- Prominently feature important functions.
- Hide more sophisticated and less frequently used functions.
    - Provide defaults.
    - Minimize screen alignment points.
    - Make common actions simple at the expense of uncommon actions being made harder.
    - Provide uniformity and consistency.

Transparency:

- Permit the user to focus on the task or job, without concern for the mechanics of the interface.
    - Workings and reminders of workings inside the computer should be invisible to the user.

Trade-Offs:

- Final design will be based on a series of trade-offs balancing often-conflicting design principles.
- People's requirements always take precedence over technical requirements.