UNIT 6: USER INTERFACES

The user interface is the point at which human users interact with a computer, website or application. The goal of effective UI is to make the user's experience easy and intuitive, requiring minimum effort on the user's part to receive the maximum desired outcome.

They include both input devices like a keyboard, mouse, trackpad, microphone, touch screen, fingerprint scanner, e-pen and camera, and output devices like monitors, speakers and printers. Devices that interact with multiple senses are called "multimedia user interfaces." For example, everyday UI uses a combination of tactile input (keyboard and mouse) and a visual and auditory output (monitor and speakers).

GUI is the acronym for graphical user interface—the interface that allows users to interact with electronic devices, such as computers, laptops, smartphones and tablets, through graphical elements. It's a valuable part of software application programming in regards to human-computer interaction, replacing text-based commands with user-friendly actions. Its goal is to present the user with decision points that are easy to find, understand and use. In other words, GUI lets you control your device with a mouse, pen or even your finger.

GUI was created because text command-line interfaces were complicated and difficult to learn. The GUI process lets you click or point to a small picture, known as an icon or widget, and open a command or function on your devices, such as tabs, buttons, scroll bars, menus, icons, pointers and windows. It is now the standard for user-centered design in software application programming.

1.1 BASIC DESIGN ISSUES

The main emphasis in the design of multimedia user interfaces is multimedia presentation. There are several issues which must be considered:

1. To determine the appropriate information content to be communicated.
2. To represent the essential characteristics of the information.
3. To represent the communicative intent.
4. To choose the proper media for information presentation.
5. To coordinate different media and assembling techniques within a presentation.
6. To provide interactive exploration of the information presented.

> "*The surface representation used by the artifact should allow the person to work with exactly the information acceptable to the task: neither more nor less.* — (**Norman**, **cognitive artifacts, 1991)**"

1.1.1 Architectural Issues

An effective presentation design should be as interactive as it is informative. The user should have the freedom to choose the direction of navigation. This should be supported by user-oriented goals, context sensitive help and selection of proper media in order to represent the information.

1.1.2 Information Characteristics for Presentation

The complete set of information characteristics makes knowledge definition and representation easier because it allows for appropriate mapping between information and presentation techniques. The information characteristics specify:

- **Types** –characterization schemes are based on ordering information. There are two types of ordered data:
  1. Coordinates vs. amount which specify points in time, space or other domains.
  2. Intervals vs. ratio, which suggests the type of comparisons meaningful among elements of coordinate and amount data types.
- **Relational Structures** – This group of characteristics refers to the way in which a relation maps among its domain sets (dependency). There are functional dependencies and non-functional dependencies. An example of a relational structure which expresses functional dependency is a bar chart. An example of a relational structure which expresses non-functional dependency is a student entry in a relational database.
- **Multi-domain Relations** –Relations can be considered across multiple domains, such as:
  1. Multiple attributes of a single object set (e.g.positions, colours, shapes, and/or sizes of a set of objects in a chart);
  2. Multiple object sets (e.g., a cluster of text and graphical symbols on a map);
  3. Multiple displays.
- **Large Data Sets –** Large datasets refers to numerous attributes of collections of heterogeneous objects (e.g. presentation of semantic networks, databases with numerous object types and attributes of technical documents for large systems etc.)

1.1.3 Presentation Functions

Presentation function is a program which displays an object (e.g. printf for a display of a character). It is important to specify the presentation function independent from presentation form, style or the information it conveys. One approach of it is the set of information-seeking goals and other is hierarchical representation of media-independent presentation goals derived from a plan-based theory of communication.

1.1.4 Presentation Design Knowledge

To design a presentation, issues like content selection, media and presentation technique selection and presentation coordination must be considered.

Content selection is the key to convey the information to the user. The information should be simple and revealing. Media selection is making choice of the media that is used to convey the content. For selecting presentation techniques, rules can be used. For eg; rules for selection methods, i.e. for supporting a user's ability to locate one of the facts in presentation. For e.g. the numerical data can be effectively presented with the help of graph while audio would be suitable for narration. Coordination can be viewed as a process of composition. Coordination needs mechanisms such as:

- encoding techniques (e.g., among graphical attributes, sentence forms, audio attributes, or between media)

- presentation objects that represent facts (e.g., coordination of the spatial and temporal arrangement of points in a chart)
- multiple displays (e.g, windows)

1.1.5 Effective Human-Computer Interaction

One of the most important issues regarding multimedia interfaces is effective human- computer interaction of the interface, i.e., user-friendliness. Here are the main issues the user interface designer should keep in mind:

1. Context;
2. Linkage to the world beyond the presentation display;
3. Evaluation of the interface with respect to other human-computer inter faces;
4. Interactive capabilities, and
5. Separability of the user interface from the application.

1.2 CLASSIFICATION OF SOFTWARE: SYSTEM SOFTWARE AND APPLICATION SOFTWARE

1.2.1 Types of System Software

Here are the important types of System Software:

- **Operating systems: -** Operating system software helps you for the effective utilization of all hardware and software components of a computer system.
- **Programming language translators: -** Transforms the instructions prepared by developers in a programming language into a form that can be interpreted or compiled and executed by a computer system.
- **Communication Software: –** Communication software allows us to transfer data and programs from one computer system to another.
- **Utility programs:** – Utility programs are a set of programs that help users in system maintenance tasks, and in performing tasks of routine nature.

1.2.2 Types of Application Software

Here, are some important types of Application Software

- **Word-processing software: -** It makes use of a computer for creating, modifying, viewing, storing, retrieving, and printing documents.
- **Spreadsheet software: -** Spreadsheet software is a numeric data-analysis tool that allows you to create a computerized ledger.
- **Database software: -** A database software is a collection of related data that is stored and retrieved according to user demand.
- **Graphics software: -** It allows computer systems for creating, editing, drawings, graphs, etc.
- **Education software: -** Education software allows a computer to be used as a learning and teaching tool.

- **Entertainment software: -** This type of app allows a computer to be used as an entertainment tool.

Here are major differences between System and Application software:

System Software	Application Software
They are designed to manage the resources of the system, like memory and process management, security, etc.	They are designed to fulfill the requirements of the user for performing specific tasks.
It is written in a low-level language like a machine or assembly language.	A high-level language is used to write Application Software.
The System Software starts running when the system is powered on and runs until the system is powered off.	The Application Software starts when the user begins, and it ends when the user stops it.
The System Software is a general-purpose software	Application Software is specific purpose software.
It is classified as a package program or customized program.	It is classified as time-sharing, resource sharing, client-server.
Installed on the computer system at the time when the operating system is installed.	Installed as per user's requirements.
Capable of running independently.	Can't run independently.
Users never interact with system software as it functions in the background.	Users interact with application software while using specific applications.
System software are independent of the application software	Application software needs system software to run.
System software is crucial for the effective functioning of a system.	Application software is not extremely important for the functioning of the system.

1.3 VIDEO AT THE USER INTERFACE

Video is actually the continuous sequence of still images such that the rate of replacement of images is 15 images per second (however for better quality 30 images per second is used). Thus the video can be manipulated using the interface that is used to manipulate the image.

The user should be allowed to navigate through the video both in the forward or backward direction possibly by the use of slider. The properties of the video like the contrast, sharpness should be adjustable and if there is audio too the user should be allowed to fine tune it. These functionalities

are not as simple to deliver because of the high data transfer rate necessary is not guaranteed by most of the hardware in current graphics systems.

1.3.1 Hardware for Visualization of Motion Pictures

Special hardware for visualization of motion pictures is available today, mostly through additional video cards. Early examples of such additional hardware are IBM-M-Motion and ActionMedia II (Intel/IBM) cards, and the Parallax, Sun and RasterOps cards. Today, these cards have become an integral part of the multimedia system.

Most motion video components integrated in a window system use the chromakey methods where an application generates a video window with a certain colour. Traditionally, this colour is a certain blue (coming from a video technique used in the TV). The window system handles, in general, the video window as a monochrome pixel graphic window, but on the device level, there is a switch which allows for the selection of the display between the standard graphics and motion video. This switch usually brings the standard graphics to the screen. If the hardware switch detects motion video, such a video window presents the video signal taken directly from a camera. Using a communication-capable multimedia system, this camera can be controlled remotely. The video data may be transmitted from the camera into a computer network and then displayed.

1.3.2 Example Camera Control Application

Remote camera control is used, for example, in surveillance applications. Another example is a microscope, remotely controlled in a telesurgery environment. We discuss below an application in which an engineer remotely controls a CIM-completion process with the help of a remote-control video camera.

Application Specification:

A camera is connected to a computer which serves as a camera server through a standardized analogue interface. The camera control occurs, for example, through a serial RS-232-C interface. The camera server sends commands such as focus, zoom and position to the camera through this serial interface. The actual control of the camera is initiated by the camera-client, which can be located remotely.

In addition to the data path for camera control, there is also a video path., i.e. the video data are digitized, compressed and sent by the camera-server to the camera-client where the engineer is located. The video image taken from the camera is displayed.

User interface:

In this case, the simplest decision would have been to use the keyboard. Fixed control functions could be assigned to individual keys. For example, the keys left, right, up, and down would move the camera in the corresponding directions.

In a window system, individual buttons can be programmed to position a camera. Pushing the buttons initiates the positioning process. The particular moment is stopped explicitly with the stop button. Another possibility to position a camera is by the pushing and releasing of a button., i.e. continuous movement of the camera follows through several consecutive ‘push’ and ‘release’ button actions.

Instead of using buttons in a window system, positioning in different access can also be done through scrollbars.

Direct Manipulation of the Video Window

In our setup we decided to use a very user-friendly variant known as direct manipulation of the video window. There are two possibilities:

1. **Absolute Positioning:** Imagine a tree in the upper right corner of the video window. The user positions the cursor on this object and double-clicks with the mouse. Now, the camera will be positioned so that the tree is the center of the video window, i.e., the camera moves in the direction of the upper right corner. This method of object pointing and activating a movement of camera is called absolute positioning. The camera control algorithm must derive the position command from:
    - the relative position of the pointer during the object activation in the video window; and,
    - the specified focal distance.
2. **Relative Positioning:** Imagine the pointer to the right of the center of the video window. By pushing the mouse button, the camera moves to the right. The relative position of the pointer with respect to the center of the video window determines the direction of the camera movement. When the mouse button is released, the camera movement stops. This kind of direct manipulation in the video window is called relative positioning. A camera can move at different speeds. A speed can be specified through the user interface as follows:
    - If the mouse has several buttons, different speeds can be assigned to each button. For example, the left mouse button could responsible for slow, accurate motion (e.g, for calibration of the camera). The right buttons could be for fast movement of the camera.
    - Instead of working with several mouse buttons, the distance of the pointer to the window center could determine the speed; the larger the distance, the faster the movement of camera.

1.4 AUDIO AT THE USER INTERFACE

Audio can be implemented at the user interface for application control. Thus, speech analysis is necessary. Speech analysis is either speaker-dependent or speaker-independent. Speaker-dependent solutions allow the input of approximately 25,000 different words with a relatively low error rate.

During audio output, the additional presentation dimension of space can be introduced using two or more separate channels to give a more natural distribution of sound. The best known example of this is *stereo*.

For example, during a conference with four participants, a fixed place is assigned to each participant. The motion video of participant L is displayed in the upper left corner of the screen. The corresponding sound of this participant is transmitted only through left speaker. Participant M is visually and acoustically located in the middle. Participant R is positioned to the right. In this

example, the conference system always activates the video window with the loudest-speaking participant. The recognition of the loudest acoustic signal can be measured over a duration of five seconds. Therefore, short, unwanted and loud signals can be compensated for.

In the case of monophony, all audio sources have the same spatial location. A listener can only properly understand the loudest audio signal. The same effect can be simulated by closing one ear. Stereophony allows listeners with bilateral hearing capabilities to hear lower intensity sounds.

The concept of the audio window allows for application independent control of audio parameters, including spatial positioning. Most current multimedia applications using audio determine the spatial positioning themselves and do not allow the user to change it. An example of such an application is the audio tool for SUN workstations. The figure below shows the user interface of this audio tool:

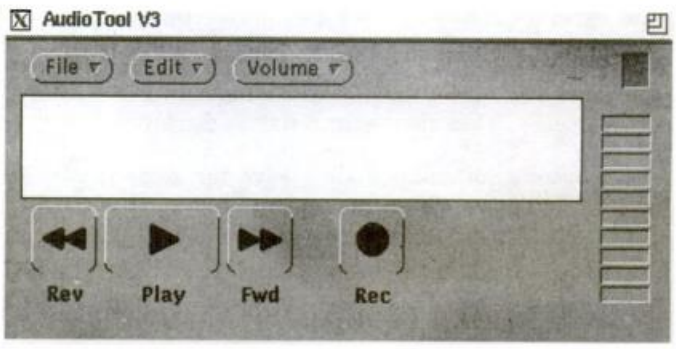

Figure 14.5: *Audio tool user interface.*

1.5 USER- FRIENDLINESS AS THE PRIMARY GOAL

User friendliness is the main property of a good user interface. The design of user-friendly graphical interface requires the consideration of many conditions. The addition of audio and video to the user interface does not simplify this process. The user-friendliness is implemented by:

1.5.1 Easy to Learn instructions

The instructions guiding the use of interface should be easy to learn. The language should be simple and graphical. The older dial phones required no time to learn. An ISDN telephone requires more time as compared to simple touch phone.

1.5.2 Presentation

The presentation, i.e., the optical image at the user interface, can have the following variants:

- Full text
- Abbreviated text
- Icons i.e. graphics

- Micons i.e. motion video

1.5.3 Dialogue Boxes

Different dialogue boxes should have a similar construction. This requirement applies to the design of :

- the buttons OK and Abort
- Joined Windows
- Other applications in the same window system

Semantically similar entry functions can be located in one dialogue box instead of several dialogue boxes.

1.5.4 Additional Design Criteria

Some additional useful hints for designing a user-friendly interface should be mentioned:

- The former of the cursor can change to visualize the current state of the system. For example, a rotating fish instead of a static pointer shows that a task is in progress.
- If time intensive tasks are performed, the progress of the task should be presented. For example, during the formatting of a disk, the amount formatted is displayed through a filling bar; during the remote retrieval of a file, the number of transmitted bytes in relation to the whole size of the file is presented. This display allows the user to evaluate the state of the task and react to it, i.e., let the task continue or cancel it. Thus, the Abort function to cancel the activity should always be present during a time-intensive task.
- A selected entry should be immediately highlighted as "work in progress" before performance actually starts. This approach ensures that no further input is given to the entry.

1.5.5 Design-specific Criteria

In addition to the above-described general criteria for the design of a user interface, the problem specific properties of the actual task need to be considered. These properties are demonstrated in our telephone service example. The telephone network and telephone-specific end-devices are provided by the telephone companies. They specify the user interface characteristics:

1. The end-device must have the basic function of dialling a number. The requirement may be that the dialling is performed using keys and that there is an alphanumeric, single- line display. This requirement provides compatibility among different phone devices. In a multimedia system, dialling with keys can be programmed, but it is not very meaningful; the main advantages of the different media are unused. To provide compatibility, a key set with corresponding user procedures should be emulated.
2. Ongoing tasks should be signalled. For example, if the call re-routing function is activated, this function should be signalled optically on the device. In the case of a telephone (computer) application, its state does not have to be displayed on the whole screen. A telephone icon can be used if no window is opened, but the application is still active.
3. A telephone device must always be operational. This requirement influences the corresponding hardware and software. If a telephone service is implemented on PCs as a

multimedia application, these devices are not always meant to be operational for 24 hours. It also cannot easily become operational when a call arrives.

4. The state of the telephone-device (i.e., telephone application) must be always visible. While working with the telephone application, it gets into different states. In different states different functions are performed. Some states imply that a function can be selected, some states imply that a function cannot be selected. The nonselective functions can be:
    - Nonexistent: The function disappears when no activation is possible.
    - Displayed: The function is displayed, but marked as deactivated and any interaction is ignored; for example, deactivated menu functions are displayed in gray, the active functions are displayed in black.
    - Overlapped: If a function is overlapped with another window which is designed as the confirmer, this function cannot be selected. First, after closing the confirmer, other input can be taken.

It is important to point out that the functions most often used are always visible in the form of a control panel. It is necessary to pick carefully which functions will belong in the control panel.

5. When a call request arrives, it must be immediately signalled (e-g., ringing).

Design of a user interface is also influenced by a specific implementation environment. For example, in addition to the primitives of the particular window system, the quality of the graphical terminal with its resolution, size and colour-map is important.

THANK YOU!!!

Assignments:

1. What are the issues which must be considered at design of multimedia user interface?
2. How can user interface be made user-friendly? Explain
3. What are the primary goals which should be considered at design of multimedia user interface?