UNIT 5
Software Tools and Interactive Devices

Unit Outcomes

- **Choose between different software tools for building interfaces.**
- **Select the appropriate devices for interaction based on the application.**

Components of UI Software

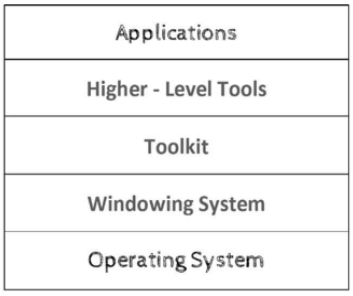
Introduction

- User interface software is most of the time large, complex, and difficult to implement, debug, and modify.
- An average of 48% of the application code is of the user interface, and that about 50% of the time is taken to implement it.
- In order to help design and implement the user interface software, special software systems and tools have been developed.
- These tools increase the productivity of the developers and many of them are commercial products today.

Components of User Interface Software

The User Interface software is divided into various layers:

- The windowing system
- The toolkit
- Higher-level tools



Components of User Interface Software

The Windowing System:

- Helps in the separation of the screen into different rectangular regions, called windows.
- Some operating systems separate windowing systems into two functional layers: Window system and Window manager.
- Window System: It is the functional or programming interface and provides procedures that allow the application to draw pictures on the screen and get input from the user.

Components of User Interface Software

- **Window manager: is the user interface and allows the end user to move windows and display the title lines, borders, and icons around the windows.**

Toolkit: contains many commonly used widgets such as menus, buttons, scroll bars, and text input fields.

Higher-level tools: Help the designer use the toolkit widgets.

Software Tool

- **A software tool is a set of computer programs that are used by developers to create, maintain, debug, or support other applications and programs.**
- **It can also be defined as a program that is employed in the development, repair, or enhancement of other programs or of hardware.**
- **A software tool can include guidelines and methodologies for interface development.**

Commonly Used Software Tools

- **Specification methods: These are methods used to specify a Graphical User Interface.**
- **Interface building tools: These are design methods that help in designing command languages, data-entry sequences, and widgets.**
- **Evaluation tools: Tools to evaluate the correctness and completeness of the programs.**

Specification Methods

Specification Methods

- The user interface design process becomes easy if the designer describes these interfaces clearly and precisely.
- There are many formal specification techniques used in software engineering to describe many aspects of software systems.
- But these techniques have not been widely applied to the problem of specifying user interfaces.

Steps in the design of a User Interface

- The designer describes and studies a variety of interfaces through a formal specification technique.
- The required user interface is chosen and before it is built, its performance is predicted by applying a Human performance model derived from empirical data.
- A prototype or mockup of the user interface can be built directly from the specification using an executable specification language.

Steps in the design of a User Interface

- This mockup can then be used to gather experimental data about the proposed user interface early in the design process.

Commonly Used Specification Methods

Grammars:

- Grammars are useful to specify textual commands or expressions that are understandable by a program.
- They provide confirmations for completeness and correctness.
  - These were earlier used with terminal-based interfaces and today are used on interactive systems.
- Grammars are useful to verify the validity of forms filled in by users on a computer. Example: Telephone book entries.

Commonly Used Specification Methods

Menu-selection and dialog-box trees:

- **These are simple structures that guide the designers in creating a selection style for many applications.**
- **The specification methods use online drawing tools to construct menu trees so that the designers and users can see the entire tree at one time.**

Commonly Used Specification Methods

Transition Diagrams:

- **Transition diagram has a set of nodes that represent system states and a set of links between the nodes that represent possible transitions.**
- **Each link is labeled with the user action that selects that link and possible computer responses.**

Commonly Used Specification Methods

State Charts:

- **It is the representation of a state machine that depicts the flow of control from one state to another.**
- **These are the extensions of conventional state-transition diagrams in terms of the notions of hierarchy, concurrency, and communication.**
- **These diagrams convert the language of state diagrams into a highly structured and economical description language.**

Interface Building Tools

Interface Building Tools

Some interface-building tools and their functionalities are as follows:

- Specification methods: are used in the design of components of the system, such as command languages, data-entry sequences, and widgets.
- Screen-transition diagrams drawn or printed on paper: help in providing an overview of the system.

These two tools allow UI architects, designers, managers, users, and software engineers to discuss and prepare a paper-based design.

Interface Mockup Tools

- These are tools used to develop a quick sketch of a GUI.
  - Example: Visual Studio .Net, etc.
- In the earlier stages of design, user interface architects create quick sketches showing multiple alternatives in the design.
- This will allow communication within the design team, and convey to the clients the look of the product
- Designers can build user-interface prototypes with multimedia construction tools, such as Dreamweaver.

Interface Mockup Tools

- User interface mockups can be created with paper and pencil, word processors, or slide show presentation software (such as Microsoft PowerPoint or Apple Keynote).

Software Engineering Tools

- These are programming tools to provide user interface management systems.
- Programmers build user interfaces with general-purpose programming languages such as Java, C#, or C++.
- But due to the lack of uniform terminology used to describe the tools and their features, choosing among them is a complex task.

The Windowing System Layer

- A windowing system is a component of a graphical user interface (GUI) in a desktop environment.
- It supports the implementation of window managers and provides basic support for graphics hardware, and pointing devices such as mice, and keyboards.
  - Example: The mouse cursor is drawn by the windowing system.
- It implements graphical primitives such as rendering fonts or drawing a line on the screen.

GUI Toolkit Layer

- GUI toolkits are user interface program libraries that provide the designers with common widgets, such as windows, scroll bars, pull-down or pop-up menus, data-entry fields, buttons, and dialog boxes.
- Sometimes a GUI tool kit library is available with a programming language. Here an experienced programmer can have great flexibility in the interface design.

GUI Toolkit Layer

- **Some toolkits are available without interactive support. These require time for programmers to gain proficiency.**
- **Example: Microsoft Windows Forms (Winforms) is a cross-platform toolkit, Apple Macintosh Toolkit (SwiftUI), and Unix X Toolkit (Xtk).**

Keyboard and Function Keys

Background

- Since 1960, there has been steady progress in processor speeds and storage capabilities and to match these the input and output devices also have advanced in their features.
- Earlier ten-character-per-second teletypes have been replaced by high-speed mega-pixel graphical displays.
- Pointing devices like mice and touchscreens have been added to free users from keyboards.

Background

- Today gestural input, two-handed input, three-dimensional pointing, voice input/output, and wearable devices have been developed to meet the needs of varied users.

Keyboards and Keypads

- For textual data entry, keyboards are used.
- Keyboard size and packaging can provide different user satisfaction and usability.
- Large keyboards are used in a professional environment whereas a small keyboard is used when a user requires to enter data and manipulate objects simultaneously.
- Very tiny keyboards and touchscreens are used where there is limited text entry such as in mobile phones.

Keys

Following key features are implemented on today's keyboards:

- The keys have concave surfaces for proper contact.
- The keys are of a matte finish to reduce glare and slipping of the finger.
- The keys use a 40-125 gram force and a displacement of 1 to 4 millimeters for better typing speeds and low error rates.
- When the key is pressed, it gives a very light click giving tactile and audible feedback to the user.

Keys

- Keys like the space bar, Enter key, Shift key and Ctrl keys should be larger than the others to allow easy and reliable access.
- Other keys such as Caps Lock and Num Lock should indicate the pressed state with an embedded light.
- The placement of the cursor-movement keys should be properly done for rapid and error-free use.

Function Keys

- There are twelve function keys F1 to F12 on the keyboard.
- These keys are used to perform different functions.
- These key presses cause the operating system to command the interpreter or an application to perform certain actions on the screen.
- On a Laptop, the Fn key must be pressed along with other keys from F1 to F12 for use.

Pointing Devices

Pointing Devices

- These devices unlike keyboards, use a pointer to point to certain items to select them.
- These are very convenient during the use of computer-assisted design tools, drawing tools, or air-traffic control systems where complex information is displayed on the screen.
- The users can avoid learning commands and typographic errors on a keyboard are reduced.

Pointing Devices

- Pointing devices are very useful for small devices and large wall displays.
- Pointing devices can be categorized based on two factors:
  - Diversity of tasks, the variety of devices, and the strategy of using them.
  - Physical attributes like type of movement (linear or rotational), the dimensionality of movement (1,2, 3…), and positioning (relative or absolute).

Pointing Tasks

Pointing devices are used for the following types of interaction tasks:

- Select: Here users choose from a set of items.
  - Example: Choosing a part in automobile design
- Position: Here users point to an item in a one, two, or three-dimensional space.
  - Example: Dragging and dropping a block of text in a figure.

Pointing Tasks

- **Orient: Here users choose a direction in a one, two, or three-dimensional space.**
  - **Example: Rotating a symbol on a screen.**
- **Path: Here the users perform a series of positioning and orientation operations.**
  - **Example: Drawing a curved line.**
- **Quantity: Here users specify a numeric value.**
  - **Example: Setting a parameter like a page number in a document.**

Pointing Tasks

- **Gesture: Users indicate to perform an action through a simple gesture.**
  - **Example: Swipe motion to turn a page forward or backward.**
- **Text: Here users can enter, move or edit the text in a two-dimensional space.**
  - **Example: Indicating the location of an insertion or deletion.**

Types of Pointing Devices

There are two basic categories of pointing devices:

- **Direct-control: These devices directly control the screen surface.**
  - **These are easy to learn and use.**
  - **Examples: LightPen, Touchscreens, Stylus**
- **Indirect-control: These devices offer indirect control away from the screen surface.**
  - **These take time to learn and use.**
  - **Examples: Mouse, Joystick, Trackball**

Speech and Auditory Interfaces

Introduction

- With the help of speech and auditory interfaces, users can interact with computers with the help of speech.
- This has become possible with the progress in hardware technology.
- But practical applications can be successful only if the cognitive load on the users and error rates are less.

Use of Auditory Interfaces

- For users with vision impairments
- For hand-free interaction
- When mobility is required
- When user's eyes are occupied
- During a condition when the keyboard cannot be used

Challenges of designing Speech Recognition Systems

- Cognitive load is more compared to pointing devices
- Interference from environmental noise
- Variation in recognition across different users and time.

Challenges of designing interfaces for Speech output

- Slow pace of speech output
- Ephemeral nature of speech
- Difficulty in scanning and searching

Technologies used by Speech Systems

- Discrete-word recognition
- Continuous-speech recognition
- Voice information systems
- Speech generation
- Non-speech auditory interface

Discrete-Word Recognition

- The devices with discrete-word recognition capability can recognize individual words spoken by a person.
- Accuracy is 90 to 98% for 100 – 10,000-word vocabulary.
- There are two types of discrete-word recognition systems: Speaker dependent and Speaker independent.
- Used to control wheelchairs, operate equipment, or use computers by people with disabilities like paralysis or unable to use their hands due to an injury.

Continuous-Speech Recognition

- These systems enable users to dictate letters and compose reports verbally for automatic transcription.
- They also enable automatic scanning and retrieval from radio and television programs, lectures, telephone calls for specific words and topics.
- Major challenges for designing such systems are recognizing the boundaries between spoken words, different accents, variable speaking rates, disturbing background noise.

Voice Information Systems

- These systems are also known as Interactive Voice Response (IVR) and use voice for communication.
- These are commonly used to provide telephone-based information about some government services and tourist sites.
- They provide voice prompts to guide users about flight timings, can help reserve tickets, and so on.
- These systems offer services at a lower cost.
- Example: Audio tours in museums and audio books

Speech Generation

- It a technology used to produce human speech artificially.
- Speech generation and digitized speech segments are used when messages are simple and short, messages are to be given at a particular time, and require an immediate response.
- This technology has been used to produce inexpensive, compact, and reliable systems for automobile navigation, internet services, utility control rooms, and games.

Speech Generation

- They are used to give spoken warnings in cockpits and control rooms.
- These systems are advantageous to users when their visual channels are overloaded, they need to move around and when the environment is too bright to see a message on a display.

Non-Speech Auditory Interfaces

Introduction

- The non-speech auditory outputs include audio tones and information presented as a combination of sound and music.
- Example: Teletypes in the earlier days used a bell tone to alert the users of an incoming message or the paper had run out.
- Non-speech audio messages can communicate to the computer user without interfering with an application
- Example: Today's computers use a range of tones to indicate warnings or the completion of a task.

Types of Sounds

There are three types of sounds created for generating different types of outputs:

- **Auditory icons: Familiar sounds**
  - **These are used to reinforce the visual experience in a graphical user interface, such as a door opening, liquid pouring or a ball bouncing in games.**

Types of Sounds

- **Earcons: Unfamiliar sounds which are to be learned to understand their meaning.**
  - **These are used to draw the attention of the users with a sharp rising tone as used in mobile devices or in a control room.**
- **Cartoonified: These are familiar sounds used in a different way to add realism and engage users in a powerful way as in cartoon animations.**