

**CMR INSTITUTE OF TECHNOLOGY**

**Department of MCA**

**Odd Semester 2016**



Internal Examination – I

**Semester : III**  
**Subject Code : 13MCA355**  
**Subject Name: Principles Of User Interface Design**

**Date:07/09/2016**  
**Time:9.00-10.30 am**  
**Max. Marks : 50**

---

**1(a).Define User Interface Design with example. 5M**

**Ans Definition :**User Interface is a part of a computer system with which a user interacts in order to undertake tasks and achieve goals.

2. User interface is the front-end application view to which user interacts in order to use the software. User can manipulate and control the software as well as hardware by means of user interface. Today, user interface is found at almost every place where digital technology exists, right from computers, mobile phones, cars, music players, airplanes, ships etc.

User interface is part of software and is designed such a way that it is expected to provide the user insight of the software. UI provides fundamental platform for human-computer interaction.

UI can be graphical, text-based, audio-video based, depending upon the underlying hardware and software combination. UI can be hardware or software or a combination of both.

**1(b).Write short notes on Usability measures. 5M**

1. Time to learn  
How long does it take for typical members of the user community to learn how to use the actions relevant to a set of tasks?
1. Speed of performance  
How long does it take to carry out the benchmark tasks?
2. Rate of errors by users  
How many and what kinds of errors do people make in carrying out the benchmark tasks?  
Although time to make and correct errors might be incorporated into the speed of performance, error handling is such a critical component of interface usage that it deserves extensive study.
3. Retention over time  
How well do users maintain their knowledge after an hour, a day, or week? Retention may be linked closely to time to learn, and frequency of use plays an important role.
4. Subjective satisfaction

5. How much did users like using various aspects of the interface? The answer can be ascertained by interview or by written surveys that include satisfaction scales and space for free-form comments.

## **2. Discuss in detail about Usability Requirements. 10M**

1. **Task analysis to ensure proper functionality**
  - Define what tasks and subtasks must be carried out
  - Pay special attention to those tasks which are only performed occasionally, common tasks are easy to define.
  - Functionality must complete, or else users will reject or underutilize the product.
2. **Reliability, Availability, Security, and Data Integrity**
  - Commands must function as specified
  - Data displayed must reflect the actual database
  - Appease the user's sense of mistrust
  - System must be error free
  - Ensure the user's privacy by protecting against unwarranted access, destruction of data, and malicious tampering
3. **Standardization, Integration, Consistency, and Portability**
  - *Standardization*: use pre-existing industry standards
  - *Integration*: product should be able to run across different software tools and packages
  - *Consistency*:
    - compatibility across different product versions
    - compatibility with related, non-computer based systems
    - use common action sequences, terms, units, colors, etc. within the program
  - *Portability*: allow for the user to convert data across multiple software and hardware environments
4. **Schedules and Budgets**

Late products can make a company ineffective or uncompetitive

## **3. Explain Usability Motivations in detail. 10M**

1. Life-critical systems
  - Air traffic control, nuclear reactors, power utilities, police & fire dispatch systems
  - High costs, reliability and effectiveness are expected
  - Length training periods are acceptable provide error-free performance
  - Subject satisfaction is less an issue due to well motivated users Retention via frequent use and practice
2. Industrial and commercial uses
  - Banking, insurance, order entry, inventory management, reservation, billing, and point-of-sales systems
  - Lower cost may sacrifice reliability

- Training is expensive, learning must be easy
- Speed and error rates are relative to cost, however speed is the supreme concern  
Subject satisfaction is fairly important to limit operator burnout
- 3. Office, home, and entertainment applications
  - Word processing, electronic mail, computer conferencing, and video game systems
  - Choosing functionality is difficult because the population has a wide range of both novice and expert users
  - Competition cause the need for low cost
- 4. Exploratory, creative, and cooperative systems
  - Database, artist toolkits, statistical packages, and scientific modeling systems
  - Benchmarks are hard to describe due to the wide array of tasks
  - With these applications, the computer should "vanish" so that the user can be absorbed in their task domain
- 5. Sociotechnical systems
  - Designers have to take into consideration the diverse levels of expertise that users with different roles have.
  - For the professional administrators and the seasoned investigators will enable rapid performance of complex procedures with visualization tools to spot unusual patterns or detect fraud in usage logs

#### **4.Explain Foley’s four level approach for designing? 10M**

- **Conceptual level:** User's mental model of the interactive system. Two mental models for image creation are paint programs that manipulate pixels and drawing programs that operate on objects. Users of paint program think in terms of sequences of actions on pixels and groups of pixels, while users of drawing programs apply operators to alter and group objects. Decisions about mental models affect each of the lower levels.
- **Semantic level:** Describes the meanings conveyed by the user's command input and by the computer's output display For example, deleting an object in a drawing program could be accomplished by undoing a recent action or by invoking a delete-object action. Either action should eliminate a single object and leave the rest untouched.
- **Syntactic level:** Defines how the units (words) that convey semantics are assembled into a complete sentence that instructs the computer to perform a certain task.
- **Lexical level:** Deals with device dependencies and with the precise mechanisms by which a user specifies the syntax

#### **5.What are the different interaction style which has to be considered using design principle? Explain each style with appropriate advantages and disadvantages. 10M**

- *Direct manipulation :*

When a clever designer can create a visual representation of the world of action, the users' tasks can be greatly simplified, because direct manipulation of familiar objects is possible. By pointing at visual representations of objects and actions, users can carry out tasks rapidly and can observe the results immediately. Keyboard entry of commands or menu choices is replaced by use of pointing devices to select from a visible set of objects and actions. Direct manipulation is appealing to novices, is easy to remember for intermittent users, and, with careful design, can be rapid for frequent users.

- *Menu selection*

In menu-selection systems, users read a list of items, select the one most appropriate to their task, and observe the effect. If the terminology and meaning of the items are understandable and distinct, users can accomplish their tasks with little learning or memorization and just a few actions. This interaction style is appropriate for novice and intermittent users and can be appealing to frequent users if the display and selection mechanisms are rapid. For designers, menu-selection systems require careful task analysis to ensure that all functions are supported conveniently and that terminology is chosen carefully and used consistently. Advanced user-interface building tools to support menu selection provide an enormous benefit by ensuring consistent screen design, validating completeness, and supporting maintenance.

- *Form filling*

When data entry is required, menu selection alone usually becomes cumbersome, and form filling (also called fill in the blanks) is appropriate. Users see a display of related fields, move a cursor among the fields, and enter data where desired. With the form filling interaction style, users must understand the field labels, know the permissible values and the data-entry method, and be capable of responding to error messages. Since knowledge of the keyboard, labels, and permissible fields is required, some training may be necessary. This interaction style is most appropriate for knowledgeable intermittent users or frequent users.

- *Command language*

For frequent users, command languages provide a strong feeling of being in control.

Users learn the syntax and can often express complex possibilities rapidly, without having to read distracting prompts. Error messages and online assistance are hard to provide. Command languages and lengthier query or programming languages are the domain of expert frequent users, who often derive great satisfaction from mastering a complex set of semantics and syntax. Powerful advantages include easy history keeping and simple macro creation.

- *Natural language*

Natural-language interaction usually provides little context for issuing the next command, frequently requires clarification dialog, and may be slower and more cumbersome than the alternatives. Still, where users are knowledgeable about a task domain whose scope is limited and where intermittent use inhibits command-language training, there exist opportunities for natural-language interfaces

### **Advantages and disadvantages of interaction styles.**

#### **Direct manipulation**

Visually presents task concepts

Allows easy learning

Allows easy retention

Allows errors to be avoided

Encourages exploration

Affords high subjective satisfaction

May be hard to program

May require graphics display and pointing *devices*

#### **Menu selection**

Shortens learning

Reduces keystrokes

Structures decision making

Permits use of dialog-management tools

Allows easy support of error handling

Presents danger of many menus

May slow frequent users

Consumes screen space

Requires rapid display rate

#### **Form fillin**

Simplifies data entry

Requires modest training

Gives convenient assistance

Permits use of form-management tools

Consumes screen space

### Command language

Is flexible

Appeals to "power" users

Supports user initiative

Allows convenient creation of user-defined macros

Has poor error handling

Requires substantial training and memorization

### Natural language

Relieves burden of learning syntax

Requires clarification dialog

May not show context

May require more keystrokes

Is unpredictable

)1

## 6. What are the eight golden rules of interface design?

### Ans Use the Eight Golden Rules of Interface Design

The following rules are mentioned to be the golden rules for GUI design

- **Strive for consistency** - Consistent sequences of actions should be required in similar situations. Identical terminology should be used in prompts, menus, and help screens. Consistent commands should be employed throughout.
- **Enable frequent users to use short-cuts** - The user's desire to reduce the number of interactions increases with the frequency of use. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.
- **Offer informative feedback** - For every operator action, there should be some system feedback. For frequent and minor actions, the response must be modest, while for infrequent and major actions, the response must be more substantial.
- **Design dialog to yield closure** - Sequences of actions should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans and options from their minds, and this indicates that the way ahead is clear to prepare for the next group of actions.
- **Offer simple error handling** - As much as possible, design the system so the user will not make a serious error. If an error is made, the system should be able to detect it and offer simple, comprehensible mechanisms for handling the error.

- **Permit easy reversal of actions** - This feature relieves anxiety, since the user knows that errors can be undone. Easy reversal of actions encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.
- **Support internal locus of control** - Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.
- **Reduce short-term memory load** - The limitation of human information processing in short-term memory requires the displays to be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.

**7. What are the guidelines available in  
(a).Getting the User's attention.**

- Intensity
- Marking
- Size
- Choice of fonts
- Inverse video
- Blinking
- Color
- Color blinking
- Audio

**7(b).Facilitating Data Entry**

1. Consistency of data-entry transactions
2. Minimal input actions by user
3. Minimal memory load on user
4. Compatibility of data entry with data display
5. flexibility for user control of data entry