* GOLDEN RULES FOR INTERFACE DESIGN :-

1.) Strive for consistency in action sequences / layouts / command use.

2.) Enable frequent users with short cuts → special key sequences, macros to perform regular / familiar functions

3) Offer informative feed back → for every user action appropriate to action's magnitude.

4.) Design dialogs to yield closure → user knows when the task is completed.

5) Permit easy reversal of actions → relieves anxiety and promotes exploration (previous state is possible)

6.) Offer error prevention / Error Handling → users should be prevented from making Errors/Mistakes and clear feed back should be given in case of errors to Recover.

7) Support Internal locus of control → user has full control of the system.

8) Reduce short term memory load → keep displaying simple, consolidate multiple page displays and provide time for learning action sequences.

*. NORMAN's PRINCIPLES :-

1.) Use both Internal and External knowledge.

2.) Simplify structure of tasks.

3) Make things / Effects visible

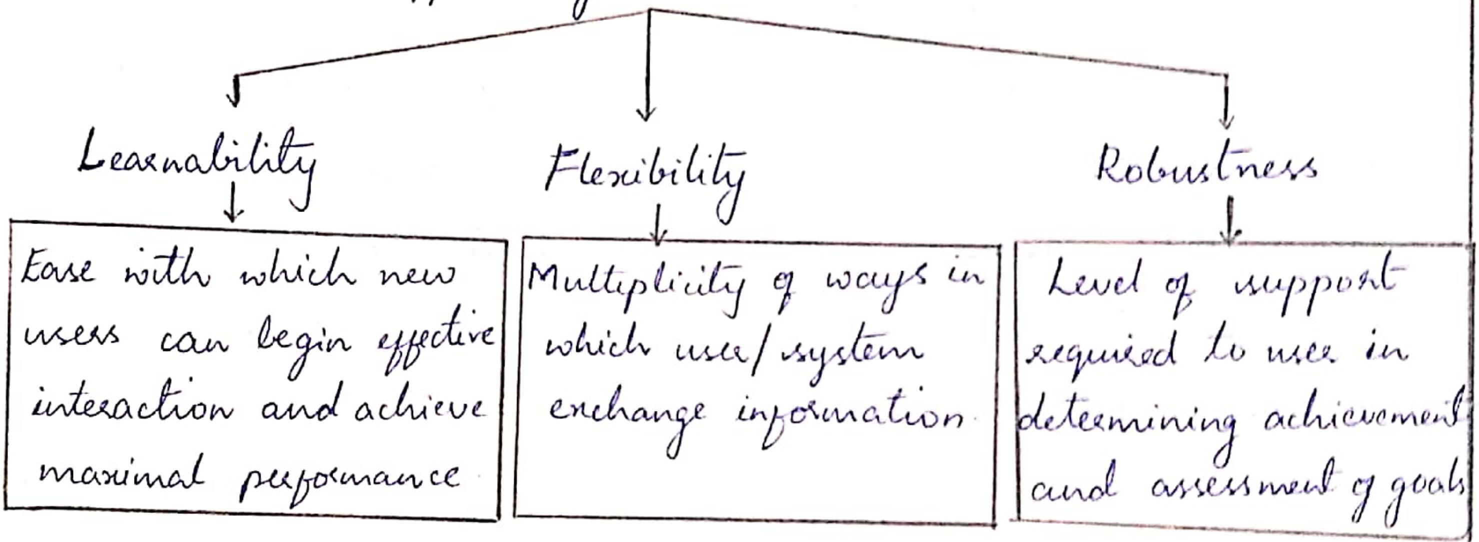
→ What system can do? and How? } should be
→ Effects of action } → explicit to user

4.) Get mappings right → user intentions should clea
map onto system controls.
What does what? How much?
Eg:- In case of controls/sliders/Bars → small/large movements mapped to small/large effects.

5) Design for Errors

6.) Standardize to defaults.

*) Principles Supporting USABILITY :-

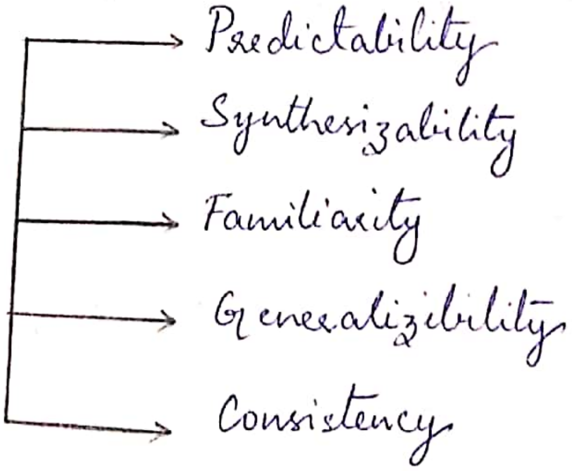


Learnability → Ease with which new users can begin effective interaction and achieve maximal performance

Flexibility → Multiplicity of ways in which user/system exchange information.

Robustness → Level of support required to user in determining achievement and assessment of goals

*) Learnability



- Predictability
- Synthesizability
- Familiarity
- Generalizibility
- Consistency

→ Predictability :- User's knowledge of interaction history should be sufficient to determine the result of future interactions.

It is an user centred concept and it is based on the deterministic behaviour from the user perspective.

*) Selection of points on drawings in microsoft applications such as word.

*) Operation visibility → availability of operations that can be performed.

→ Synthesizability :- Ability of the user to assess the effects of past operations on current state.

*) "Honesty" is the ability of user interface to provide an observable / Informative account of change.

*) "Immediate" means notifications can occur without delay or atleast eventually.

Eg :- Visual desktop interfaces or Command language.

Consider "file move from one directory to another"

→ Visual setup → Honest → Immediately

→ Command line → Honest → Eventually

↓

Eg :- (Linux) → "mv" command and then "ls" in both directories.

Other Examples :-

→ In Apple Mac machines (earlier versions), in case of creation of new folder with in a folder → visual effects were not immediately honest.

→ Similarly global search / replace functions of word processor are not immediately honest.

→ Familiarity :- Correlation between user's existing kn
and knowledge required for effective interaction. This
nothing but "Guessability of the system".
Eg :- Typewriter to Word processor transition

"Affordances" → how objects can be manipulated

Affordance → enhances → Familiarity

→ Generalizibility :- It is a form of consistency and specific to general cases transition support. It is similar to principle of Mathematical Induction / Inductive Reasoning.

Eg :- 1.) In case of Graphical Applications
circle → constrained ellipse
square → constrained rectangle

2.) Cut / copy / paste operations → same effects on multiple windows / Applications.

→ Consistency :- Likeness in Behaviour arising from similar situations / task objectives.

Eg :- 1.) Consistent keyboard layout
QWERTY / DVORAK keyboard layouts

2.) Colour coded warning panel in aircrafts
red → immediate recovery required
amber → Eventual recovery required
Green → Advisory recovery required.

exibility
- → Dialog Initiation
- → Multi Threading
- → Task Migratability
- → Substitutivity
- → Customizability

→ Dialog Imitiation :- System / User preemptive.
(a) (b)

a) System initiates actions towards the user.

b) User initiates actions towards the system.

Dialog initiation facilitates Balanced trade off.

Eg :- Shared Documents (Google Docs) → Multiple users editing documents – "consistent feature" → system preemptive dialogs.

→ Multi Threading :- Interaction to support more than 1 task at a time.

Multithreading
- Concurrent
  - ↓ Simultaneous communication of information pertaining to separate tasks.
- Interleaved
  - ↓ Temporal overlap between separate tasks but stimulus at a given instant i.e., dialog is restricted to a single task.

→ Multimodality of a dialog is related to multithreading of a dialog.

Eg :- Beep when editing a file (Arrival of New message in Inbox)

Eg :- Windowing system → Multi-threaded dialog.

→ Interleaved amongst number of overlapping tasks.

→ Task Migratability :- Transfer of control for executi
tasks between system and user.

*) Transfer of control across both.

Eg:- 1) Spell checking → system /user control.

2.) In safety critical applications → system /user control is essential as it is a matter of life /death.

→ Substitutivity :- Alternate forms for action sequences.

*) Representation multiplicity ⇒ flexibility for state rendering

*) Equal opportunity for input /output levels

"System /user not premptive"

Eg:- 1) Margin setting in MS word / multiple ways

2) Temperature graph → Digital thermometer
(trends) (values)

3) Excel — spread sheet → Formula.

→ Customizability :- Modifiability of user Interface. Customizability is nothing but automated modifications of system based on knowledge (level) of user.

Customizability

① Adaptability ② Adaptivity

① Adaptability → User's ability to adjust the form of input/output

Eg:- Consider the position of soft buttons → limited in operation

"Structure of Interaction unchanged"

aptivity :- Automatic customization of user interface by the system

*) Based on user Expertise

*) knowledge of HCI patterns / Behaviour history.

* **Robustness**
- → Observability / Browsability / Reachability
- → Recoverability
- → Persistence
- → Responsiveness
- → Task Conformance.

→ Observability :- Evaluate Internal state of the system by means of perceivable representation at interface

Observability
- → Browsability
- → Defaults
- → Reachability
- → Persistence
- → Operation visibility

i) Browsability → Explore current state via limited view at interface.

*) No side effects

*) Passive with respect to system state.

ii) Defaults :- Error prevention mechanism.

Defaults are defined within system or acquired during installation.

Defaults
- ↙ static
- ↘ Dynamic

Dynamic defaults – Evolve during the session

Static defaults – Not evolved during the session

In case of static defaults, default values are adopted based on user behaviour.

iii) Reachability → Possibility of Navigation through the observable system states.

iv.) Persistence → Duration of Effect of a communication act and the ability of user to make use of that effect.

Eg:- Beep on receipt of mails – Reminder during other interleaved operations.

v) Operation visibility :- Honest / Immediate Effects of action sequences.

→ Recoverability → Ability to reach a desired goal after recognition of some error in a previous interaction.

Recoverability

↙ Backward ↓ Undo effects, back to earlier constant state

↘ Forward ↓ Acceptance of current state and negotiation from that state to a desired state

Eg:- ^z / undo button in word processors.

→ Principle of "commensurate effort" → in worst case, it takes as many actions to reach a safe state as it took to reach error state.

SPONSIVENESS -: Rate of communication between the system and the user.

Response time is the duration of time needed by the system to express state changes to the user.

*) In case of short durations → Instantaneous response times are desirable.

*) Feedback during intensive computations depends on response time.

*) Stability of response time is also vital.

→ Task Conformance :- Task completeness addresses coverage, task adequacy which means user's understanding of the task.

* Standards / Guidelines :-

→ Principles - Abstract Design Rules with "High generality and Low authority"

→ Standards - Specific Design Rules "High in authority and low in application"

→ Guidelines - low in authority and high in application.

*) Standards for Interactive system Design are usually set by National / International bodies to ensure compliance with a set of design rules.

*) Standards for hardware are based on understanding of physiology or Ergonomics / Human factors.

This results in fixed / readily adaptable rules to design hardware.

Software standards are based on psychology/cognitive science: "less well formed / Evolving"

*) Hardware is more difficult to change and expensive than software. Hardware changes less frequently than software. Standards are stable for hardware than software
∴ Standards are more suited for hardware.

A few STANDARDS -:

1.) Arrangement of Displays → Vertical Grouping
→ Engine parameters → ordering as per importance of parameter

2.) Division of display area → Input area
→ Output area
→ Operations area.

A few GUIDELINES -: 1.) Data Entry 2.) Data Display 3.) Sequence controls 4.) User guidance 5.) Data transmission 6.) Data Protection

*) Data Entry → "Position Designation Guideline"
Eg:- 1) Distinctive cursor → cursor shape/animation/multiple

2) Dialog styles → Q/A forms, menus,
function keys, command language, graphic set ...