Introduction to Software Engineering

- User Interface Design

>	UI design is an iterative process involving close liaisons between users and
>	The 3 core activities in this process are:

- designers.

—	User analysis. Understand what the users will do with the system;
—	System prototyping. Develop a series of prototypes for experiment;
—	Interface evaluation. Experiment with these prototypes with users.

- Four different models occur in HCI design:

1.	The design model expresses the software design.
2.	The user model describes the profile of the end users. (i.e., novices vs. experts, cultural
3.	The user's model is the end users' perception of the system.
4.	The system image is the external manifestation of the system (look and feel +

- background, etc.) documentation etc.)
- Principle
- Description
- User familiarity
- Use terms and concepts familiar to the user.

Consistency	same way. Commands and menus should have the
Minimal surprise	should be able to predict the operation of comparable

- Comparable operations should be activated in the same format, etc. If a command operates in a known way, the user commands.
- Feedback
- Provide the user with visual and auditory feedback, maintaining two-way communication.
- Characteristic
- Description
- Windows
- Multiple windows allow different information to be displayed simultaneously on the user's screen.

Icons	applications), but they may also stand for processes
Pointing	command choices from a menu or indicating items of

- Usually icons represent files (including folders and (e.g., printer drivers).
- Menus
- Menus bundle and organize commands (eliminating the need for a command language). A pointing device such as a mouse is used for interest in a window. Graphical elements can be commands on the same
- Advantages

>	They are easy to learn and use.
>	The user may switch attention between tasks and applications.
>	Fast, full-screen interaction is possible with immediate access to the entire screen
>	A GUI is not automatically a good interface

- — Users without experience can learn to use the system quickly. Problems — Many software systems are never used due to poor UI design — A poorly designed UI can cause a user to make catastrophic errors Advantages

>	Users feel in control and are less likely to be intimidated by the system
>	User learning time is relatively short
>	Users get immediate feedback on their actions
>	mistakes can be quickly detected and corrected
>	Finding the right user metaphor may be difficult

>	**Users feel in control and are less likely to be intimidated by the system**
>	It can be hard to navigate efficiently in a large information space.
>	It can be complex to program and demanding to execute

- Problems
- Advantages

>	**Users don’t need to remember**
>	Typing effort is minimal
>	User errors are trapped by the
>	Context-dependent help can be
>	Actions involving logical conjunction (and) or disjunction (or) are
>	If there are many choices, some menu structuring facility must be

- command names interface provided (based on the current menu selection) Problems awkward to represent
- Scrolling menus

>	**The menu can be scrolled to reveal additional choices**
>	Not practical if there is a very large number of choices
>	Selecting a menu item causes the menu to be replaced by a sub-menu
>	A menu selection causes another menu to be revealed
>	When a menu item is selected, a control panel pops-up with further options

- Hierarchical menus Walking menus Associated control panels