At the end of this lecture students should have some understanding of:

- what are interfaces and dialogues
- how we specify interface and form design
- the different types of interaction - command language, menu, form and icon-based
- issues concerning data entry
- ways of controlling user access
- the use of graphical user interfaces (GUI)

Designing Interfaces and Dialogues

The process of defining the manner in which humans and computers exchange information

Analogous to a conversation between 2 people

Interface and dialogue design is critical for successful information systems

"...to the user the interface is the system"

Should provide a uniform structure for finding, viewing, and invoking different components of an information system

Use prototyping as with forms and reports

Design specifications - also as with forms and reports

- Narrative overview
- Interface/dialogue design
- Testing and usability information

Narrative overview

- interface name, user, task, system, environment

Interface/dialogue design

- form/report designs, dialogue sequence diagram

Testing and usability assessment

- testing objectives, procedures and results
- time to learn, speed of performance, rate of errors, retention over time, user satisfaction and other perceptions

Common interface methods are:

- Command language interaction
- Menu interaction
- Form interaction
- Icon/object-based interaction
Users enter explicit commands to invoke operations
- users must therefore remember syntax and semantics
- MSDOS, UNIX commands
  - copy A:\myfile.doc to B:\myfile.doc
- Complex applications such as word processors and spreadsheets may have many commands
  - <CTRL-P> print
  - <CTRL-S> save
  - <CTRL-C> copy

Users select from a list of system options and a specific operation is executed
- Menu interaction is still the most widely used interaction method
- Menus can differ greatly in their design and capability
  - simple single menu
  - hierarchy of menus
  - pop-up menus
  - drop-down menus
  - graphic menus

Menu Interaction

Simple Single Menu

Menu Hierarchy

Drop-Down Menu

Guidelines for Menu Design
- Wording
  - meaningful title, clear, unambiguous command verbs
- Organisation
  - consistent, eg. related options should be grouped together
- Length
  - use sub-menus to break up very long menus
- Selection
  - methods should be clear and consistent
- Highlighting
  - should be minimised and used to convey selected options (marked) or unavailable options (dimmed)
Form Interaction
- Users fill in ‘blanks’ on a form when they interact with the system
- Screen format is similar to paper-based forms
- Effective for both input and presentation of data
- Form interaction is common on many applications, the World Wide Web
- Form-based programming languages
  - VB, Delphi, CGI(?)

Icon/Object-Based Interaction
- Often based on metaphor
- Icons are graphic symbols that look like the processing option they represent
- Operations are selected by pointing at and clicking on the desired option
- Easily understood and take up little screen space
- Many icons appear in the form of buttons or controls

Common Interaction Devices
- Keyboard
- Mouse
- Joystick
- Trackball
- Touch screen
- Light pen
- Graphics tablet
- Voice

Structuring Data Entry
- Entry
  - never require data to be inserted that is already known
- Defaults
  - always provide defaults where appropriate
- Units
  - make clear units of measure
- Captioning
  - always provide a caption near fields to explain meaning

Structuring Data Entry
- Format
  - provide format details (decimal points, dollar signs, dates …)
- Justify
  - automatically justify data entries - numbers right justified; text left justified
- Help
  - always provide context sensitive help (use F1)

Validation of Input data
- Tests are usually handled by DBMS or by programs:
  - Type
    - proper type of data (eg. numeric, alpha …)
  - Combinations
    - do values in several fields combine in a sensible way
  - Missing data
    - eg. must have quantity for each customer order
Validation of Input data

- These tests are usually handled either by a DBMS or by programs:
  - Range
    - are data within proper range of values?
  - Size
    - correct number of characters?
  - Values
    - does value come from a standard set of values?

Other Interface Design Features

- Provide feedback
  - give status information to keep users informed
    - eg. Please wait while ...
  - give prompting cues
    - eg. Enter customer account no.
  - give meaningful error and warning messages
    - avoid jargon, be specific
- Provide help
  - for help, concepts, procedures, messages, menus, function keys, commands, etc.

Controlling User Access

- Mechanisms to control user access include
  - views
    - particular users see subset of DB in the interface
  - authorisation rules
    - restrictions to access and actions on data
  - encryption procedures
    - encryption and decoding procedures are necessary for highly sensitive data
  - authentication schemes
    - access restrictions controlled by passwords, biometric devices and smart cards

Designing Dialogues

- 3 major steps:
  - 1 Designing the dialogue sequence
    - based on how the user wishes to use the system
  - 2 Build a prototype
    - use tools such as CASE tools, Visual Basic, Access
  - 3 Assess Usability
    - use same approach as with forms and reports - time to learn, speed of performance, rate of errors, retention over time, subjective satisfaction

Graphical User Interfaces

- Are becoming increasingly powerful
- Are becoming something of a standard
- All previous design guidelines apply, together with several additional ones
  - organisational requirements, contextual issues
- Typical GUI environments include Macintosh and Microsoft Windows - these come with extensive guidelines and consistency standards

References
