Interface Design

**What is a user interface?**

* Describes how users interact with a computer system
  + Gets input from the user
  + Provides output to the user

**When does user interface design occur?**

* Design phase
* User-centered design
* Usually done with prototyping

**Principles**

* Layout
  + Each area may be further subdivided
  + Each area is self contained
  + Areas should have a natural intuitive flow
    - Left to right
    - Top to bottom
* Content Awareness
  + Must intuitively answer:
    - Where am i?
    - What am I supposed to be doing here?
  + Applies to sub areas within a form or window
    - Address information is grouped together
    - Report information is grouped together
* Aesthetics
  + Interfaces should be inviting to use
  + Readable
  + Proper colors and fonts
  + Less is more
  + White space is important
  + Information density is proportional to user experience
    - Novice = less
    - Experts = more
* User experience
  + Ease of learning
    - Important for novice users
    - Relevant to systems with large number of users
  + Ease of use
    - Important for expert users
    - Important for specialized systems
  + Sometimes ease of learning and use go hand in hand
  + Types of computer users
    - Expert
      * Experienced in systems and will spend time using specific applications
    - Novice
      * Inexperienced user who will use a computer on a less frequent basis
  + Guidelines
    - Clearly label all controls
    - Select easily understood images to represent actions
    - Provide clear and concise on-screen instructions
    - Make it easy to return to any level in the menu structure
    - Provide shortcuts for experienced users
    - Provide auto complete functionality (like google search)
* Consistency
  + All parts of the system work in the same way
  + Key areas are:
    - Navigation controls
    - Terminology
  + Make it simple so users can predict what is going to happen
* Minimal user effort
  + Interfaces should be designed to minimize effort needed to accomplish tasks
  + Three click rule

**Design Process**

* Interface design prototyping
  + Mock-ups or simulations of computer screens forms and reports
  + Common approaches
    - Storyboard
    - Windows layout diagram
    - HTML prototype
    - Language prototype
* Prototyping the user interface
  + Initial prototype is designed form requirements
    - Mock-ups or simulations
  + User review prototype design and either accept the design or request changes
  + If changes are requested, the cycle repeats until client is happy

**Prototyping**

* Make sure there is a standard
  + Button placement
  + Colour scheme
  + Location of error/help text
* First pass is to verify that all necessary attributes are on the screen
* Final version should be exercised by user
  + Demonstrate how to get help/instructions
  + Demonstrate how security is handled
  + Demonstrate error handling

**Forms**

* A business document that contains pre-defined data and may include some additional areas to fill out
* Is an input to the system.

**Reports**

* A business document that contains only predefined data
* An output of the system
* Used for reading and viewing data

**Creating Forms and Reports**

* User-focused activity
* Follows prototyping approach
* Requirements determination:
  + Gain understanding of the intended user
  + Understand the task objectives
  + Who will use the form/report?
  + When will it be used
  + What is the purpose?

**Commandments of UI Design**

* Understand user and their tasks
* Involve the user in interface design
* Test the system on actual users
* Practice iterative design

**Navigation Design Basic Principles**

* Prevent mistakes
* Simplify recovery from mistakes
* Use consistent grammar order

**Navigation Guidelines**

* The user must always be aware of what to do next
* Tell user what the system expects now
* Notify the user that the data was entered correctly or not
* Explain to the user why there’s a delay in processing
* Tell user what tasks are completed and not
* Display the information in the same general area

**Message Types**

* Error Message
* Confirmation message
* Acknowledgement message
* Delay message

**Message Guidelines**

* Messages should be displayed long enough for a user to view it
* Alert users on lengthy processing times
* Provide onscreen progress reports
* Default values for fields and answers to be entered by user should be specified

**Error Handling Guidelines**

* Anticipate errors users might make
* Do not let user proceed without correcting an error
  + Highlight the error
  + All user to make corrections without retyping the whole thing
* If user does something catastrophic prevent any further input and inform them to call IT
* Provide an undo key or menu choice to reverse an action

**Help Guidelines**

* Ensure help is always available
* Title every help screen
* Provide user-selected help
* Provide context-sensitive help (offer help for task in progress)
* Provide direct route for users to return to point where help was requested

**Tone and Terminology Guidelines**

* **Tone**
  + Use simple, grammatically correct sentences
  + Don’t be funny, cute or condescending!
* **Terminology**
  + Don’t user computer jargon
  + Avoid abbr.
  + Use simple terms
  + Be consistent

**Input Design**

* Best principles
  + Online versus batch processing
  + Capture data at the source
  + Minimize key strokes
* Types of input
  + Free form
    - Text box
    - Number box
  + Selection box
    - Check box (yes or no or multiple options available
    - Radio button (when 1 item can be selected)
    - List box (drop downs, combo box)
    - Sliders

|  |  |
| --- | --- |
| **Validation Type** | **When to Use** |
| **Completeness** | **Ensures all required data has been entered** |
| **Format/datatype** | **Ensures data are of the right type (eg. Numeric) and the right format (eg. A date is month, day, year)** |
| **Range** | **Ensures numeric data are within a correct min and max. (eg. Rejecting birthdates prior to 1880.)** |
| **Check digit/self-checking digit** | **When numeric equations are used to validate information, such as when checking credit card numbers, SIN** |
| **Consistency** | **Ensure that a field makes sense in relation to another field. Eg. when the user enters both a birth date and a date of marriage (birth < marriage). When a form asks user to confirm password (form must contain values for both password and password confirmation, and they must match)** |
| **Limit** | **When there is a min or max, but not both** |
| **Sequence** | **A predetermined sequence exists (regular expression). Eg. Postal code** |