Design - not just software

Design for software and beyond

• Why are some everyday things difficult to use?
• How can we apply Don Norman's principles to the design of computer interfaces?

Design for software and beyond

• Make things visible
• Provide a good conceptual model
  - Affordance
  - Mapping
  - Constraints
  - Feedback

Don Norman's design principle: Visibility

The correct parts must be visible and they must convey the correct message.

Visibility problems occur when clues are lacking or exist in excess.

Just by looking, the user should know:
  - State of the system
  - Possible actions

Don't violate these principles to make something "look good"!

Don Norman's design principle: Visibility

Visibility problems occur when clues are lacking or exist in excess.

Larson's dog effect works with Software

Thank you for registering! We appreciate your business. To activate your software, you will be sent an email key. After you have received the key then you will be able to click here and you can then proceed with the activation process.
Don Norman's design principles

Provide a Good Conceptual Model (Metaphor)
• A good conceptual model allows us to predict the effects of our actions
• Without a good model we operate blindly!
  – Simply follow rules without understanding a reason
  – No understanding of cause or effect
  – No recourse when something breaks
Name several good conceptual models used in software.
Apple's desk top model
iBook’s book shelf model.

Don Norman's design principles

Affordances
• The affordances of an object determine, naturally, how it can be used.
  – Button affords pushing
  – Handle affords grasping
  – Chair affords sitting
  – Knob affords turning
• Just by looking at the object, a user should know how to use it.
  – Example: A door with handles means to pull.

Affordances and Constraints
Example of poor door design!
People get trapped between the two doors!
You need to pull on one side and push on the other.
Handles afford pulling.
Handles provide pulling naturally.
Using a flat plate would constrain the user to push.
Constraints limit the ways in which something can be used.
Constraints can be
- Physical
- Semantic
- Cultural
- Logical

Don Norman's design principles

Mapping
• Controls and displays should exploit natural mapping.
• Natural mapping takes advantage of physical analogies and cultural standards.
  – Physical: Steering wheel
  – Cultural: red means stop, green means go

Don Norman's design principles

Poor design due to poor Mapping
Where do you plug in the mouse?
Don Norman's design principles

Poor design due to poor Mapping
Which control turns on which burner?

Feedback
• Feedback is sending back to the user information about what action has actually been done.
• Visibility of the effects of the operation tell you if something worked correctly.
• Systems should be designed to provide adequate feedback to the users to ensure they know what to do next in their tasks.

Feedback Examples
• Telephone button press tones – Telephone clicks
• Rice cooker goes “bing!”
• Clicker on your turn signal
• Animated icon while waiting for a web page to load.

Name some examples of poor feedback.

Where do these go wrong?

Mapping  Visibility  Feedback  Affordances

Win NT Dialog  How to turn on this stove?

This handle unfastens the seat from the floor.

Norman's design principles in software

• Visibility
  – Visibility of the tasks the interface supports
  – Communication of system state / mode

• Affordance
  – If it looks like a button, it can be pressed.
  – If it is underlined, it can be clicked (web).

• Mapping
  – Clicking on a particular interface element produces expected effect, e.g., Open should be under the File menu.

• Constraints
  – Constraining search criteria, e.g., graying out menu items that don’t apply in a particular context.

• Feedback
  – Providing clear and immediate feedback for each user action.
Don Norman's design principles

Human Computer Interface (HCI) – research area in CS
- designing effective Graphical User Interfaces (GUI)

References