# Design for Interaction

Principles of Interaction Design

# Contents

- Usability goals
- The user experience
- Design principles

# Aims of Interaction Design

- There are certain basic aims involved in the creation of any interactive system
  - The system should be <u>usable</u> – in other words, it should meet the needs of the users when they are interacting with it
  - The system should provide a positive <u>user experience</u>
- Considering these issues allows us to identify key goals for any design

# Usability Goals

- Based around the following criteria:
  - Effectiveness
  - Efficiency
  - Safety
  - Utility
  - Learnability
  - Memorability

# Effectiveness

- The ability of the system to enable whatever user activity it is supposed to allow
  - How well it does its job
- Examples:
  - Does a DVD menu system allow a viewer to easily access all the content and features
  - Does an online flight booking system allow valid purchases to be made correctly

# Efficiency

- The way in which interactive system design enables user "productivity"
- Being able to access desired content or functionality with minimal effort or complications
- Example:
  - Amazon.com's one-click ordering option

# Safety

- The employment of design principles which ensure users can't make critical errors or lose data or functionality

- Examples:
  - Greying out inapplicable options in a menu
  - Not placing critical functions (eg. 'quit') adjacent to other, routinely used options
  - Using 'confirm' dialogues for critical functions
  - Allowing (multiple) 'undo'

# Utility

- The range of <u>relevant</u> functionality offered by an interactive system
- In theory, the more features a system has, the more usable it will become
  - Users who have all their needs met by the system don't need to look elsewhere
- This assumes that the functions are easily accessible and well integrated

# Learnability

- The ease with which a user can learn to interact <u>fluently</u> with the system
- This will depend in part on the way the system is intended to be used
  - Systems with a higher utility can get away with relatively poor learnability

# Memorability

- How easy it is to recall how to use a system after some time away from it

- Especially important when a system is likely to be used infrequently

- Basic principles of memorability:
  - The system should appear to be organised in a clear and logical manner
  - Icons, menu titles, etc should be meaningful and grouped into sensible categories

# Learnability and Memorability

- Obviously a link between the two
- The importance of both is task-dependent
- How learnable and memorable must these tasks be made on a digital TV recorder?
  - Playing a pre-recorded programme
  - Setting it a day in advance to record a single programme
  - Setting it to record all programmes in a series
  - Rescanning the available channels

# The User Experience

- Usability goals allow us to identify designs which carry out a required task

- Any product type will have many usable design options, each of which will generate a different <u>user experience</u>

- We should aim to create user experiences which suit the application
  - Identify a range of user experience goals

# User Experience Goals

- Possible experience goals:
  - Satisfying
  - Fun
  - Entertaining
  - Helpful
  - Aesthetically pleasing
  - Motivating
  - Emotionally fulfilling

# Notes on the User Experience

- Not all these terms will apply to all possible systems

- Some might be incompatible or unsuitable

- For example, a flight training simulator shouldn't be designed to be aesthetically pleasing
  - Should mimic the real experience as closely as possible

# Usability Design Principles

- From our list of usability and user experience goals, we can identify basic interaction design principles
  - Visibility
  - Feedback
  - Constraints
  - Mapping
  - Consistency

# Visibility

- Available functions and features need to be clearly visible

- "Hiding" important functions, or leaving them unlabelled will frustrate users
  - "Mystery meat" navigation
  - http://www.webpagesthatsuck.com/mysterymeatnavigation.html

- Furthermore, the positions and relationships between interface elements should aid user actions
  - Example: car controls

# Feedback

- User actions should produce corresponding activity from the system

- Feedback should be relevant to the type of user action

- It should also be <u>immediate</u>

- Feedback can add to visibility – eg. tool tips

# Constraints

- Limit the accessible options to only those which are relevant to the current task
- One example is the greying-out of irrelevant menu items in a software application
  - Increases safety by preventing user errors
  - Can provide clues and guidance as to how a task should be carried out
- Relies on the user understanding the nature of – and reason for – the constraint

# Mapping

- Functionality should, where possible, mimic the real-world equivalent actions
- An example is a typical interface for a media player (MP3 player, PVR, etc)
  - Controls for play, rewind and fast forward will typically have a logical ordering
  - << > >> (not: > << >> )
- Similarly, forward and back buttons on a web browser mimic the printed page

# Consistency

- Systems used for similar tasks (or the same tasks in different contexts) should function in similar ways

- A classic example is in the integration of software suites such as Microsoft Office or Adobe Creative Suite

- Similarly, an operating system such as Windows formalises the idea of consistency

# Example: MS Office