

cis3.5
introduction to multi-media computing
spring 2009
class session # 1.1
introduction

topics:

- introduction to the course
- what is multi-media computing?
- introduction to user interface design

instructor:

- Prof Elizabeth Sklar, sklar@sci.brooklyn.cuny.edu, AIM screen name: *agentprof*

course web page:

- <http://www.sci.brooklyn.cuny.edu/~sklar/cis3.5>

introduction to the course

- broad introduction to topics in Multi-Media Computing (MMC), including: web design, game design, data visualization, simulation, animation and robotics
- discussions about a broad range of subjects, such as: multimedia hardware and software; human interface design and input using multi-media devices; graphical and other forms of output to multi-media devices; computer-based sound editing; agent-based programming for simulations and robotics; uses of multi-media in industry
- emphasis on design and creation of a range of artifacts, including: web pages with cascading style sheets; interactive, graphical web-based programs; simple computer games, movies and narratives
- format consists of a mixture of lecture and laboratory class sessions
- the following topics will be covered in 4 units:
 - (I) Introduction to Web Programming and Web Design
 - (II) Interactive Web Programming and Data Visualization
 - (III) Game Programming, Narrative and Movie Making
 - (IV) Agent-based Programming, Simulation and Robotics

course structure

- **4 units**
- each unit has:
 - 2-3 lectures
 - 3-5 labs
 - 1 project
- the labs will be hands-on sessions using either workstations in the library (384L), laptops in the classroom (4411N) or workstations and/or laptops in the multi-media lab (5301N)
- the projects will consist of:
 - computer-based component
 - written/descriptive component
 - (sometimes) oral component
- your grade = 4 projects (15% each) + midterm (10%) + file exam (30%) = 100%

about me

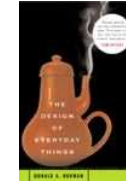
- undergrad: Barnard, CS major/English minor, BA 1985
- 10 years of industry experience working as a scientific and business programmer
 - MIT Lincoln Laboratory (<http://www.ll.mit.edu/>)
- grad school: Brandeis University, MA 1997, PhD 2000
- previous teaching:
 - Monash University, Melbourne, Australia (Summer 2000)
 - University of Melbourne, Melbourne, Australia (Summer 2000)
 - Boston College, Massachusetts (Fall 2000–Spring 2001)
 - Columbia University (Fall 2001–Spring 2005)
 - Brooklyn College, Fall 2005– ...
- research interests center around educational technologies:
 - multi-media learning environments and educational robotics
 - interactive learning systems
 - multi-agent simulation and artificial intelligence (AI)

what is multi-media computing (MMC)?

- MMC represents a fusion of multiple types of data sources used to acquire, process, transmit, store, and utilize information.
- MMC produces a whole that is greater than the sum of its parts.
- MMC takes computing and communication beyond traditional text-dominated documents and plays an increasingly significant role in our lives.
- Multi-media content includes: digital images, movies, music and animations.
- Multi-media content appears on many devices, including: laptops, cell phones, ipods.
- Multi-media collaboration requires exchange of multi-media content, stored on distributed/remote peers/servers.
- Multi-media functionality includes disseminating, broadcasting, streaming and downloading music files, movies, images, graphics and other multi-media content.
- MMC is used by diverse communities including: computing specialists, multimedia technology developers, artists, health-care providers and environmentalists.
- *information source:* <http://www.sigmm.org>

introduction to user interface design

- Psychologist Donald Norman used the term “affordance” in his book *The Design of Everyday Things* (1990)



- The term can be applied to both physical and virtual objects.
- Here, we are interested in virtual objects that appear on a computer screen.
- The important aspect is what the user *perceives*.
- Can we design an interface where the user perceives that actions s/he takes will enable the results s/he desires?
i.e., where the interface *affords* the actions intended by the designer?

design principles

- Norman says: “In the world of design, what matters is:
 1. If the desired controls can be perceived
 - 1.a. In an easy-to-use design, if they can both readily be perceived and interpreted
 2. If the desired actions can be discovered
 - 2.a. Whether standard conventions are obeyed”
- Four principles for interface design:
 1. Follow conventional usage, both in the choice of images and the allowable interactions.
 2. Use words to describe the desired action (e.g., “click here” or use labels in front of perceived objects).
 3. Use metaphor.
 4. Follow a coherent conceptual model so that once part of the interface is learned, the same principles apply to other parts.
- Some devices/interfaces *force* functionality; e.g., starting a car requires a key
- Design for error—expect that the user will make errors; what do you think the user might do wrong? Plan for these errors: eliminate them; if you can't, then handle them!

eight rules of interface design

1. Be consistent
2. Aim for universal usability
3. Provide helpful feedback for every user action
4. Provide closure with dialogs
5. Prevent errors where possible and otherwise handle errors elegantly
6. Allow reversal of actions
7. Make users feel in control (“internal locus of control”)
8. Limit memory load

source: Andrew Johnson (2006) <http://www.ev1.uic.edu/aej/422/>

recommendations

- screen real estate:
 - how much space will things take up?
 - where will your user have the mouse?
 - where will they want to move the mouse?
- use of color:
 - color can help but be aware of color blindness
 - limit the number of colors
 - use color to group things together (that you want to group together)
 - use color to support a task (e.g., brightness makes things easy to find)
 - remember color conventions (e.g., red, yellow, green)
- grouping of objects
 - group related objects together
 - visual hierarchy can reflect object hierarchy
 - visual relationships: position, size, “weight”

recommendations (continued)

- balance (use it!)
- clutter (avoid it!)
- ask yourself: where does the eye naturally go?
- use symmetry to ensure balance
- use alignment to establish visual relationships between objects
- consider human optical adjustment
 - use the “squint” test to see if things you want to stand out, do stand out
 - don’t make the eye wander all over back and forth across the screen)
- use “negative space” or “white space”
 - this refers to space that is not what you want the user to look at, but space that helps separate items and clarify the visual elements in a design

things to avoid

- things that don’t work the way you expect them to
- different things that are too similar
- things that are hard to see
- things that don’t work well together
- things that get in the way
- things that are hard to handle
- things that are hard to remember
- things that don’t fit
- displays that look like controls
- incompatible/unexpected/unnatural mapping of controls to devices

to do

- in class:
 - fill out pre-semester survey and give it to me before you leave today
 - work on today’s lab
- at home:
 - check out the class web page:
<http://www.sci.brooklyn.cuny.edu/~sklar/cis3.5>
 - finish today’s lab