

Bringing Design to Software, ch1

Jon Kolko
Savannah College of Art & Design

IDUS317 - Interactive Product Design | 1

Overview

- Mitchell Kapor
- Big Ideas
- Pleasurable Software
- Design and Programming Education
- Applied

IDUS317 - Interactive Product Design | 2

About the Author



Mitchell Kapor
[<http://www.kei.com/homepages/mkapor/>]

- Lotus 1-2-3
- MIT Media Lab (cognitive psychology, AI)
- His weblog
[<http://blogs.osafoundation.org/mitch/>]

IDUS317 - Interactive Product Design | 3

A Manifesto

This article is called "A Software Design **Manifesto**".

"A public declaration of principles, policies, or intentions, especially of a political nature" (dictionary.com).

Why would it be important to issue a *manifesto* concerning software design?

IDUS317 - Interactive Product Design | 4

Big Ideas

*"Despite the enormous outward success of personal computers, the daily experience of using computers far too often is still fraught with **difficulty, pain, and barriers** for most people, which means that the revolution ... has not yet succeeded" (3).*

- Lack of advocacy
- Lack of understanding
- Lack of **design**.

What does design mean in this context (p3)?

IDUS317 - Interactive Product Design | 5

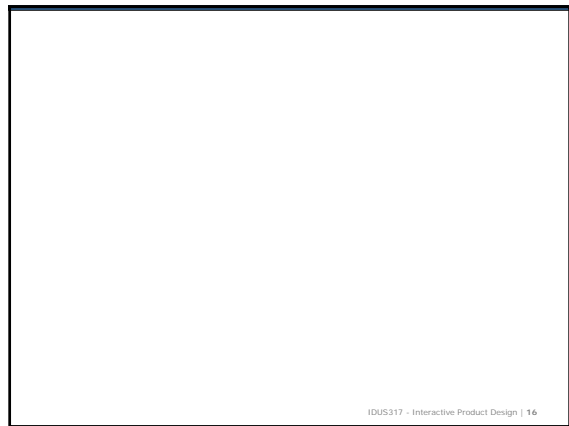
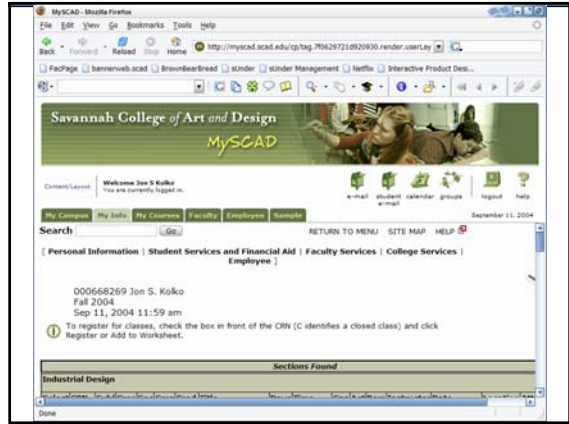
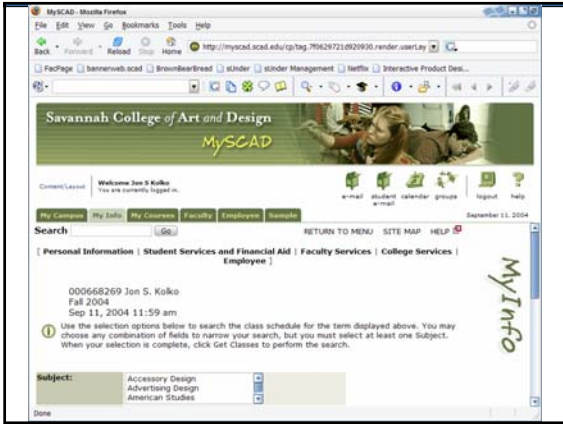
The Case for Design

"Design disciplines are concerned with making artifacts for human use ... One of the main reasons most computer software is so abysmal is that it's not designed at all, but merely engineered" (4).

- **Firmness:** A program should not have any bugs
- **Commodity:** A program should be suitable for the purposes for which it was intended
- **Delight:** The experience of using the program should be a pleasurable one

IDUS317 - Interactive Product Design | 6





Why not?

Why isn't software as pleasurable as a romp on the beach?

IDUS317 - Interactive Product Design | 18

On Training Designers

"In architecture, the study of design begins with the fundamental principles and techniques of architectural representation and composition ... many people who think of themselves as working on the design of software simply lack the technical grounding to be an effective participant in the overall process" (6-7).

What is the overall point Kapor is making for programmers?

What is the overall point Kapor is making for designers?

Is he correct?

IDUS317 - Interactive Product Design | 19

1990 Was a Long Time Ago.

Kapor wrote this in 1990.

Is it still accurate?

Why or Why Not?

IDUS317 - Interactive Product Design | 20

Two Approaches to Education

Teaching Design to Programmers

1. Understanding of logic, math
2. Data structures, software architecture, formal programming principles
3. Coding (specific language)
4. Development in a business setting
5. *User Interface fundamentals*

Teaching Programming to Designers

1. Visual Design, composition, drawing, painting
2. Color theory, semantics and meaning, design theory
3. User centered research
4. Design in a business setting
5. *Programming fundamentals*

Which is better?

Which are we doing here at SCAD?

Is this good or bad?

IDUS317 - Interactive Product Design | 21

Applied?

How can you apply Kapor's Manifesto to the current assignment?

IDUS317 - Interactive Product Design | 22

Summary

- Mitchell Kapor
- Big Ideas
- Pleasurable Software
- Design and Programming Education
- Applied

IDUS317 - Interactive Product Design | 23