

CS G256

May 11, 2004

Lecture Plan

- Homework #1 — analysis
- Homework #2 — discussion
- GUI Usability:
 - Design Process
 - Principles
 - Bloopers

Analysis of homework

#1

- Presentation
- Confidentiality
- Writing Quality

Presentation

- Treat every assignment as if it is a finished work product
- No need for a cover page on a 3-page assignment!
- Every page should have:
 - Your name
 - A page number

Confidentiality

- Some students printed the name of their employer or clients, others didn't.
- Q: When is there an obligation of confidentiality?
- A: When you have exposure to privileged information

Tips on Writing

- Don't raise questions that you don't answer.
- Explain the setting: kind of organization, operating systems, etc.
- Don't excessively quote
- Don't spend too much time on the "lessons learned" — the lessons should be obvious from the context!

More Writing Tips

- Avoid slang
- It's safer to be formal than to be lax
- Don't use acronyms without defining them.

Writing Quality

The *quality* of your written communication matters just as much as technical accuracy!



Northeastern University
Writing Center

<http://www.dac.neu.edu/owl/>

Who do we tutor?

- Undergraduate / graduate students from all academic disciplines
- Speakers of other languages and native speakers of English
- Weak writers, average writers, strong writers

How can we help?

- Brainstorming, idea generation
- Planning, outlining, organizing
- Reading with and against the grain
- Researching, critiquing
- Developing thesis / arguments
- Drafting, revising
- Coaching on grammar, punctuation, style

Issues raised

Issues raised ... I

- Infection of Windows XP machine by MSBlaster
- Unauthorized use of a webproxy at a bank
 - The proxy circumvented the anti-virus
 - But perhaps the proxy was being used accidentally? It was in the PAC file...

Issues raised ... 2

- Downstream Liability for an Internet Connection
- Running a worm (offer_letter.doc.exe) received by email
- Physical Security
 - Laptop theft
 - Dumpster diving

Issues raised ... 3

- Back door left in a program by a previous programmer
 - It got in there without any code auditing!
 - The hole was fixed, but the customer and users were never notified.
 - What were the responsibilities of the company? Of the programmer?

Issues raised ... 4

- Case of cheating on an exam, resolved by forensic analysis of the submitted problems.

HCI-SEC issues

- Why is it so hard to install updates?
 - One MSBlaster victim had downloaded the patches but never installed them --- thought the question was an advertisement!
 - One SGI user had to download 50 patches --- total time 8 hours (?)
- How do you know what a computer is really doing?

Other HCI-SEC Issues?

Designing Usable Interfaces

- What is the computer interface?
 - Command line?
 - Windows/MacOS Dominance
 - PalmOS
 - Speech & gestures?

The Design Cycle

- Task Analysis
- Iterative Design:
 - Design
 - Prototype
 - Evaluate
 - Repeat
- Keep the customer in the picture!

Task Analysis

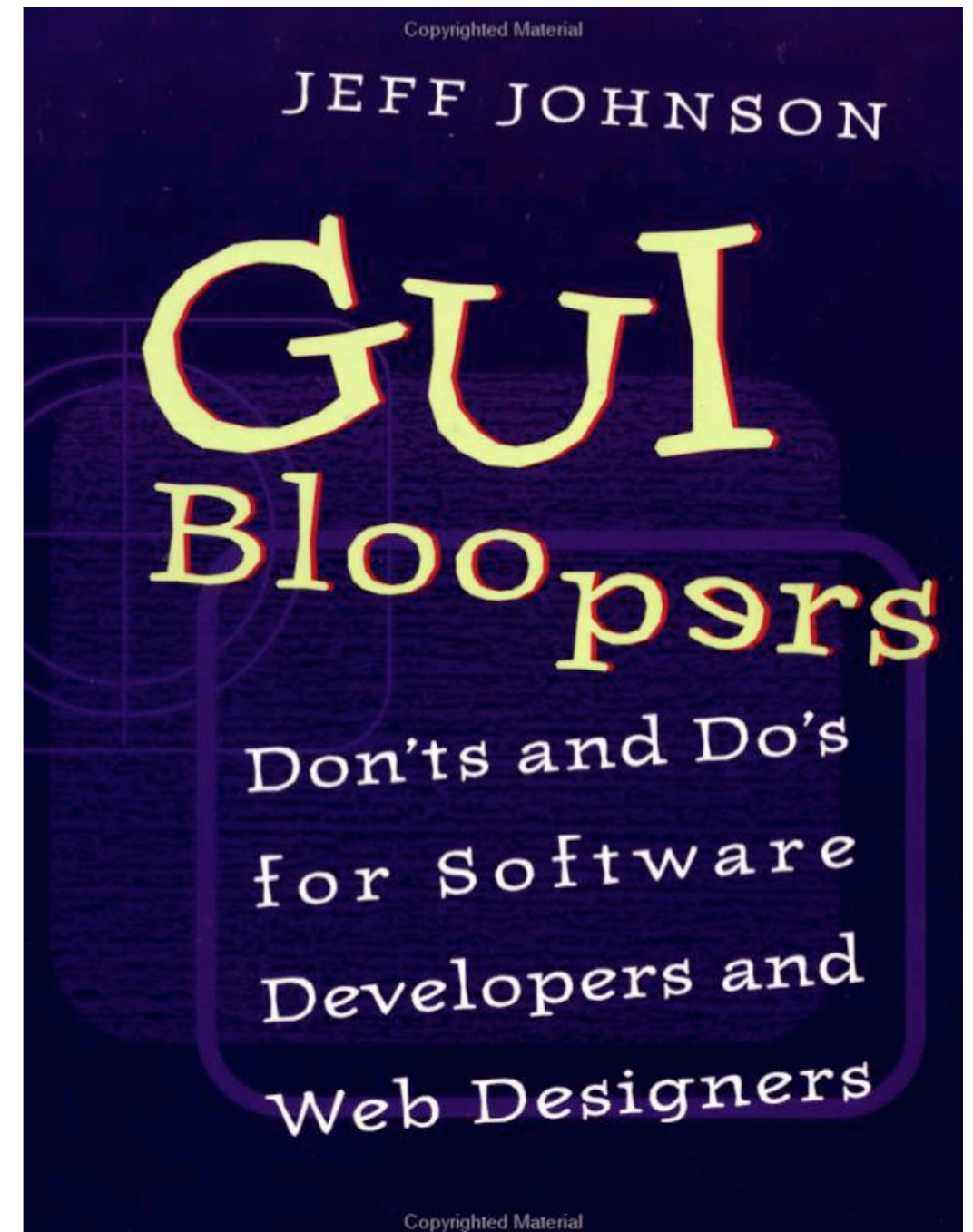
- Observe existing work practices
- Create scenarios
- Create “customers”
 - Sally in accounting
 - Bob the new user
- Discuss ideas with end-users
- Show prototypes; try out ideas before committing to software

Rapid Prototyping

- Build a mock-up
- Low-cost techniques:
 - paper!
 - Adobe Illustrator / Photoshop
- Cheap interfaces:
 - GUI builder
 - Flash

Designing usable interfaces

Jeff Johnson, *GUI Bloopers: Don't and Do's for Software Developers and Web Designers*, Morgan Kaufmann, 2000



- Principle 1: Focus on the users and their tasks, not the technology
 - For whom is this product being designed?
 - What is the product for?
 - What problems do the users have now?
 - What are the skills and knowledge of the users?
 - How do users conceptualize and work with their data?

- Principle 2: Consider function first, presentation later
 - Does not mean “worry about the user interface later!”
 - Develop a conceptual model
 - Keep it as simple as possible, but no simpler
 - Develop a lexicon (***)

- Principle 3: Conform to the users' view of the task
 - Strive for naturalness
 - Use the user's vocabulary, not your own
 - Keep program internals inside the program

- Principle 4: Don't complicate the user's task
 - Common tasks should be easy
 - Don't give users extra problems to solve
 - Example: converting a file format from TIFF to JPG for web publishing
 - Example: Installing program "A" in order to install program "B"
 - Example: Looking up information on one screen to type it on another

- Principle 5: Promote Learning
 - Think “outside-in,” not “inside-out”
 - The user wants to solve a problem, not learn how to use your program!
 - Be careful of ambiguity
 - “He saw the woman with the telescope”
 - Icons that don’t make sense
 - Be consistent so there is something to learn!

- Principle 6: Deliver information, not just data
 - Design displays carefully
 - The screen belongs to the user
 - Preserve display inertia
- Principle 7: Design for responsiveness
- Principle 8: Try it out on users, then fix it!

Rob Miller on UIs

- User interface strongly affects perception of software
 - Usable software sells better
 - Unusable web sites are abandoned
- Perception is sometimes superficial
 - Users blame themselves for UI failings
 - People who make buying decisions are not always end-users

RCM: User Interfaces are Hard to Design

- You are not the user
 - Most software engineering is about communicating with other programmers
 - UI is about communicating with users
- The user is always right
 - Consistent problems are the system's fault
- ... but the user is not always right
 - user's aren't designers

UI's are half the game:

- Myers & Rosson, “Survey on user interface programming”, CHI ‘92
- User Interfaces account for 50% of:
 - Design time
 - Implementation time
 - Maintenance time
 - Code Size
- (probably more now!)

Usability: What is it?

- “I know it when I see it.”
- Interfaces we *enjoy* using (satisfaction)
- Interfaces we are *fast* at using (efficiency)
- Interfaces that we can *use without asking for help* (*learnability*)
- Interfaces that we can use *accurately* (*errors*)
- Interfaces we can use after time (*memorability*)

UI Hall Of Shames

- http://www.rha.com/ui_hall_of_shame.htm
- <http://pixelcentric.net/x-shame/>