

# CS3724 Human-computer Interaction

## Wrap Up and Making It All Work

Copyright © 2004 H. Rex Hartson and Deborah  
Hix.

# Topics



- Review
- Selling these new techniques
- Cost justification
- Getting started
- Parting words

# Wrap Up

- *We have addressed how to design the content of quality interaction, and a process by which usability can be ensured in user interaction*
- Apply interaction design *guidelines*
- Use an iterative, evaluation-centered *usability engineering life cycle* for user interaction development
- Participate in *systems analysis*, including user, needs, task, and functional analyses

# Wrap Up

- What you can now do:
  - Perform conceptual and detailed *design*
  - Establish *usability specifications*
  - Build *rapid prototypes*
  - Perform *formative usability evaluation*
  - Iteratively refine an interaction design
  - Know how to get started with these new ideas

# Wrap Up

- Your biggest challenge may be:
  - Not technical!
  - Selling this to management
    - By necessity, the interaction development process has changed from linear to iterative, which in turn changes at least:
      - Control
      - Scheduling
      - Organizational roles
      - Territoriality



# Wrap Up

- Project Management
  - Communication, skills
  - Test facilities, tools
- What we've presented is the basis for controllability, accountability, and quantitative methods that are so important (rightfully) to management

# Wrap Up

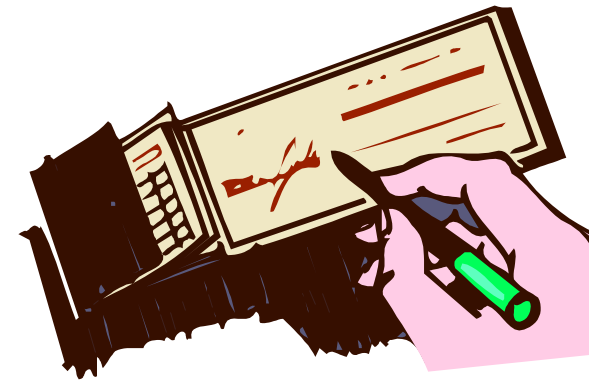
- Selling these techniques to management
  - They may not be aware that there is a problem
  - They may view these techniques as a solution to a non-existent problem
    - “The product is selling well”
    - “We’re getting lots of interested inquiries”
    - “Users don’t complain”
  - “We’ve never done it this way before”
  - Seek out corporate mission statement and show usability engineering supports
  - Remember the “personware” factor

# Wrap Up

- Successful interaction designs are being developed using these techniques, because they've been shown to work!
- Resources needed: Minimum of 15% of entire development effort!
- *“You have to keep running just to stay in the same place!”*

# Cost Justification of Usability

- Can we afford to include usability engineering in our system development process?



- Answer: Usability engineering does not add overall cost, for two reasons
  - Costs are added only to a limited part of total development process
  - Usability saves on many other costs

# Cost Justification of Usability



- Added costs are confined
  - Reality: Interaction development process must be iterative—*cannot* get it right the first time
  - But interaction development is *small* part of overall system development
  - Rest of development – user interface software and other application software – is not necessarily iterative



# Cost Justification of Usability

- Poor usability is costly; good usability is all about saving costs
    - Costs of hardware and software vs. costs of “personware”
    - Usability is about good business, not just about “being nice”
    - Costs of development vs. cost of operation
      - Development costs are mostly one time; operational costs accrue for years
      - Cost/benefit scope must be broad enough to include usage, as well as training, help desk support, etc.
- Scope problem: one group pays for development cost and another

# Cost Justification of Usability

- Development savings from usability in process
  - High software maintenance costs – trying to get it right *after* release
  - Implementation Costs
- Usage savings; even more significant if users are your employees!
  - Save operational productivity costs
  - Save user training costs



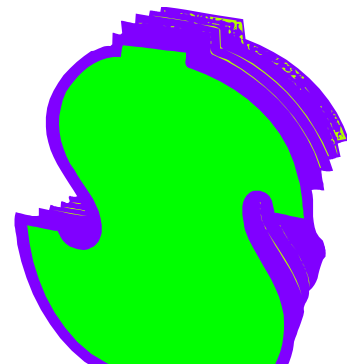
# Cost Justification of Usability



- Save costs of user errors
- Save costs of database errors
- Save costs of help desk and user support operations
- Save intangible costs of employee dissatisfaction
- ***Point: Not more resources to ensure usability, but different resources with different distribution during life cycle***

# Cost Justification of Usability

- Beyond cost savings: In the e-commerce world of the Web, good usability can mean increased revenue!
  - Can market your company as having a focus on usability; competitive edge
  - Huge need to avoid releasing something that will embarrass you and the company (despite the pressure of “Internet time”)

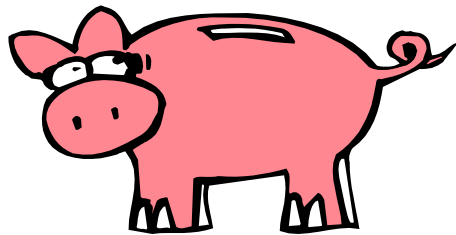


# Cost Justification: A Simple Example

- For a large distributed system:
  - Users: 75,000
  - Average transactions/users a day: 20
  - Transactions/day: 1,500,000
  - User time per transaction: 5-20 minutes
  - Average time saved per transaction, due to improved usability: 30 seconds
  - Average fully-loaded hourly rate: \$25.00



# Cost Justification: A Simple Example



- Saved per year
  - = 75,000 users \* 20 trans/user-day \* .5 min/trans \* 230 days/yr \* \$25/hr \* 1hr/60min
  - = \$71,875,000.00
- Other savings: user training, help desk
- Regardless of what usability engineering cost for this product, payback is enormous

# Cost Justification of Usability

- But won't it be nice when we no longer have to justify "costs" of usability?
- When have you heard anyone ask: Can we afford costs of designing data structures, implementing algorithms, doing quality assurance, etc.....?!

# Getting Started

- Some ideas for selling these techniques to management:

- ***Start small***

- Try the process on a small part of a project
- Try a few usability specifications
- Set up a small usability lab somewhere, anywhere – and use it
- Develop at least a minimal customized style guide



# Getting Started

- Tell management exactly what you intend to try and hope to accomplish, and within what time frame
- Expect some rough spots in initial stages
- Get appropriate resources lined up
  - Get buy-in from management

# Getting Started

- Get at least one person with appropriate skills on the user interface development team, and give them a title, responsibility, and authority
- Give appropriate training to team members
- Get commitment from team members to try these new techniques
- Find someone you can apprentice with
- Get consulting help when needed, especially during start-up

# Getting Started

- Professional preparation
  - Go to appropriate conferences – e.g., Computer-Human Interaction (CHI); Human Factors and Ergonomics Society (HFES); User Interface Software and Technology (UIST); National Institutes of Standards & Technology (NIST)
  - Subscribe to HCI publications
  - Join Usability Professionals' Association (UPA)

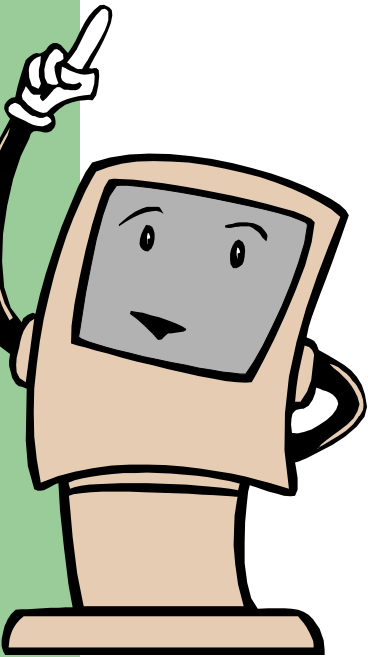
# Getting Started

- Join Special Interest Group on CHI (SIGCHI) – local and/or national
- Start a “brown bag” user interface lunch bunch
- Try the process all the way through once
- Generate a failure story
- Better: Generate a success story
  - E.g., Video-clips for “before and after”



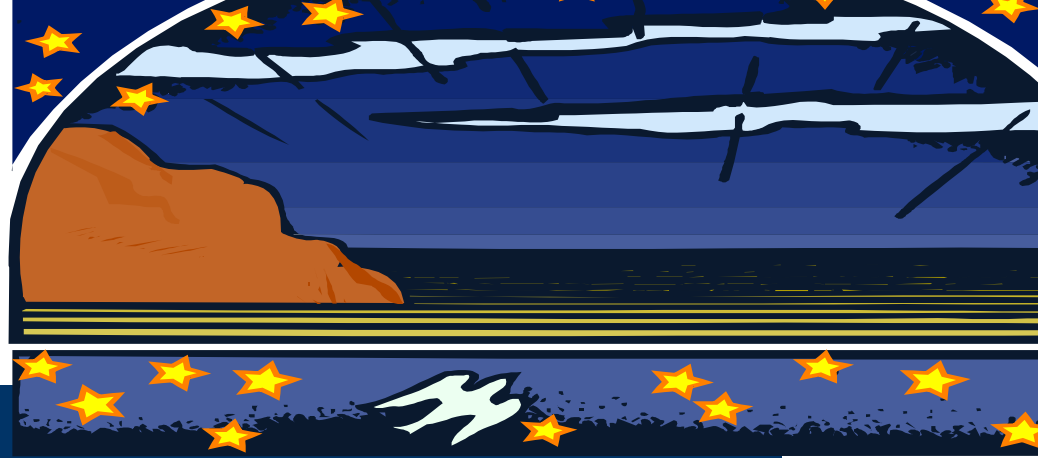
**Success**

# Parting Words



- Encourage focus on the process, rather than just the product
  - Make customized process guide
  - Operationalize the process organization-wide
- Ensure usability “by practice” rather than “by decree” or “by politics”
- Characteristics needed by user interaction/interface developers:
  - Dedication – to the cause of quality interfaces
  - Daring – to do things differently

# Parting Words



- Congratulations, you made it!!!!

