Introduction & Course Overview

CS490f

Prof. James A. Landay
University of Washington
Autumn 2006

September 29, 2006

UI Hall of Fame or Hall of Shame?

Hall of Shame!

- Doesn’t help user accomplish their task
  - why did they come to the site?
- Takes too long
  - most visitors will leave & never return
- May be valid for entertainment, art, or branding sites

Hall of Shame!

- Page setup for printing in IE5
- Problems
  - codes for header & footer information
  - requires recall!
  - want recognition
  - no equivalent GUI!
  - help is the way to find out, but not obvious

Hall of Fame or Shame?

- Page setup for printing in IE5
Introduction & Course Overview

CS490f

Prof. James A. Landay
University of Washington
Autumn 2006

September 29, 2006

Outline

• Who are we?
• HCI introduction
• Course overview & schedule
• Introductions

Who are we?

• James Landay
  – Associate Professor in CSE at the University of Washington
  – formerly professor in EECS at UC Berkeley
  – spent last 3 years as Director of Intel Research Seattle (ubicomp lab)
  – Ph.D. in CS from Carnegie Mellon ‘96
  – HCI w/ focus on informal input (pens, speech, etc.),
  Web design (tools, patterns, etc.) & Ubiquitous Computing
  – founded NetRaker, leader in Web experience management
  – Now subsidiary of KeyNote Systems
  – Co-authored The Design of Sites with D. van Duyne & J. Hong

• Kate Everitt
  – Ph.D. student in CSE
  – BSc in Computing & Info Science from Queen’s University
  – MS in CS from UC Berkeley
  – HCI w/ focus on computer support cooperative work & speech UIs

Human-Computer Interaction (HCI)

• Human
  – the end-user of a program
  – the others in the organization
• Computer
  – the machine the program runs on
  – often split between clients & servers
• Interaction
  – the user tells the computer what they want
  – the computer communicates results

HCI Approach to UI Design

Factors Influence Each Other

“People change their knowledge as they perform, i.e., they learn”
User Interfaces (UIs)

• Part of application that allows people
  – to interact with computer
  – to carry out their task

• User vs. Customer vs. Client
  – user is a term only used by 2 industries → bad!
  – customer – person who will use the product you build
  – client – the company who is paying you to build it

HCI = design, prototyping, evaluation, & implementation of UIs

Why is HCI Important?

• Major part of work for “real” programs
  – approximately 50%

• Bad user interfaces cost
  – money
    • 5% satisfaction → up to 85% profits
    • finding problems early makes them easier to fix
  – reputation of organization (e.g., brand loyalty)
  – lives (Therac-25)

• User interfaces hard to get right
  – people are unpredictable
  – intuition of designers often wrong

Who Builds UIs?

• A team of specialists (ideally)
  – graphic designers
  – interaction / interface designers
  – information architects
  – technical writers
  – marketers
  – test engineers
  – usability engineers
  – software engineers
  – users

How to Design and Build UIs

• UI Development process

  • Usability goals
  • User-centered design
  • Task analysis & contextual inquiry
  • Rapid prototyping
  • Evaluation
  • Programming

User Interface Development Process

Iteration

At every stage!
Design

• Design is driven by requirements
  – what the artifact is for
  – not how it is to be implemented
  – e.g., PDA not as important as “mobile” app.
• A design represents the artifact
  – for UIs these representations include:
    • screen sketches or storyboards
    • flow diagrams/outline showing task structure
    • executable prototypes
      – representations simplify

Web Design Representations

Site Maps

Storyboards

Schematics

Mock-ups

Usability

According to the ISO:
The effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments

• This does not mean you have to create a “dry” design or something that is only good for novices – it all depends on your goals

Usability Goals

• Set goals early & later use to measure progress
• Goals often have tradeoffs, so prioritize
• Example goals
  – Learnable
    • faster the 2nd time & so on
  – Memorable
    • from session to session
  – Flexible
    • multiple ways to accomplish tasks
  – Efficient
    • perform tasks quickly
  – Robust
    • minimal error rates
  – Option to recover
  – Pleasing
    • high user satisfaction
  – Fun

User-centered Design

“Know thy User”

• Cognitive abilities
  – perception
  – physical manipulation
  – memory
• Organizational / job abilities
• Keep users involved throughout
  – developers working with target users
  – think of the world in users terms
  – not technology-centered/feature driven

Task Analysis & Contextual Inquiry

• Observe existing work practices
• Create examples and scenarios of actual use
• “Try-out” new ideas before building software
Rapid Prototyping

- Build a mock-up of design so you can test
- Low fidelity techniques
  - paper sketches
  - cut, copy, paste
- Interactive prototyping tools
  - HTML, Visual Basic, Flash, DENIM, etc.
- UI builders
  - Visual Studio .NET, JBuilder...

Evaluation

- Test with real users (participants)
  - w/ interactive prototype
  - low-fi with paper “computer”
- Build models
  - GOMS
- Low-cost techniques
  - expert evaluation
  - walkthroughs
  - online testing

Goals of the Course

1) Learn to design, prototype, & evaluate UIs
   - the needs & tasks of prospective users
   - cognitive/perceptual constraints that affect design
   - technology & techniques used to prototype UIs
   - techniques for evaluating a user interface design
   - importance of iterative design for usability
   - how to work together on a team project
   - communicate your results to a group
     - key to your future success

2) Understand where technology is going & what UIs of the future might be like

Course Format

- Interactive lectures
- Quarter long project & homeworks
- Readings
- All material is online
  - slides, exercises, readings, schedule
  - http://www.cs.washington.edu/cs490f
- Have fun & participate!

How CSE490f Fits into CS Curriculum

- Most courses for learning technology
  - compilers, operating systems, databases, etc.
- CSE490f concerned w/ design & evaluation
  - technology as a tool to evaluate via prototyping
  - skills will become very important upon graduation
    - complex systems, large teams
    - don’t look for large immediate impact in other CS courses

Project Description

- Each of you will propose an interface idea
  - fixing something you don’t like or a new idea
- Groups
  - 4 students to a group
  - work with students w/ different skills/interests
  - groups meet with teaching staff every 2 weeks
  - industrial mentors may also meet with teams
- Cumulative
  - apply several HCI methods to a single interface
Project Process Overview

- Project proposal (individual) due Tuesday
- Break-up into groups next Thursday
- Project task analysis & “sketches”
  - i.e., rough proposals that can & will change
  - based on field work with ESM tool on phone
- In class presentations & critiques
- Low fidelity prototyping & user testing
- In class presentations & critiques
- Rapid prototype using tools & user test
- Heuristic evaluations (individual)
- Heuristic evaluation summary
- Final presentations & project fair with industry guests

What is the 2nd quarter Class?

- Takes up where this course stops
- Focus on
  - executable prototypes
  - UI toolkits & implementation
  - advanced user testing
  - even more project focused

Project Examples (cont.)

- SiteSketch
  - web page design
  - sketch-based

SiteSketch

Project Examples (cont.)

- Clothes Shopper
  - online shopping
  - knows your prefs & sizes

Clothes Shopper
Project Examples (cont.)

• Electronic book reader
  – take advantage of all the online texts on the net

Electronic Book Reader

Project Examples (cont.)

• Nutrition tracker

Nutrition Tracker

Project Examples (cont.)

• cULzine
  – recipe tool for the home

cULzine
Project Examples (cont.)

- Read WWW over phone
  - find structure in pages & build voice menus
  - navigation problem
  - cache common paths & reorder?
- PDA brainstorming tool
  - small portable computers in a group meeting (say Palm Pilots)

Project Examples (cont.)

- Runner’s training log
  - input daily workouts
  - reports
  - reminders
- PDA for shopping
  - scan in UPC & tells you whether a good price
- Home entertainment control - “no more remotes”

Total Entertainment Control

PDA Baseball Scorekeeper

PalmStock
Otto: Location-based Photos

Bus Buddy

Ubiquitous Computing Project Themes

• Location-enhanced computing
  – phones that are aware of their location
  – past examples include car navigation, Trippin’, finding nearby restaurants, etc

• Activity-based computing
  – applications that use inference of human physical activity to enhance our lives
  – helping care for an elder
  – helping people stay fit
    • exercise & nutrition

• Ubiquitous RFID
  – tags & readers

Administrivia

• Registration
  – limited by room and project constraints to 32
  – appeal email to me if not enrolled (due today at 5 PM)
  – tell us why you should be in the course
    • background, interests, what you can contribute
  – will email admits by Monday at 5 PM

• Roll
  • James’ office hours
    – Wed. 9:30-10:30 AM (642 Allen Center)
    – Mon. 3-4 PM online (send Kate Yahoo/MS/Google ID)
    – email katherine@cs.washington.edu for appointments at other times

Administrivia (cont.)

• Teaching assistants
  – Katherine Everitt
    • last name at cs.washington.edu
    • O.H.: TBA in 409 Allen Center

• Discussion sections
  – TBD – please respond to Kate’s email
  – new material will be covered in discussion
   attend
Books

- *The Design of Sites* by van Duyne, Landay, & Hong
  - I’ll give you copies of the 4-5 chapters we will use
- We will also hand out other papers, give you web links, & refer to lecture slides
- Recommended textbooks
  - Human-Computer Interaction by Alan Dix, et. al., 3rd edition, 2003
  - order from Amazon.com (link off class web page)
- Other recommended books on web page

Assignments (tentative)

- Individual
  - 3 written + one talk each
- Group
  - 4 written assignments
    - 3 presentation/demos with the write-ups
  - all group work handed in on Web (group web site)

Grading

- A combination of
  - midterm (15%)
  - final (25%)
  - individual assignments (15%)
  - group project (40%)
    - demos/presentation (group component)
    - project write-ups and exercises
    - ratings given by other team members & class
  - in class participation (5%)
- No curve

Tidbits

- Late Policy
  - no lates on group assignments
  - individual assignments lose one letter grade/day
- Cheating policy
  - will get you an F in the course
  - more than once can get you dismissed
- More information (syllabus/schedule/slides)
  - http://www.cs.washington.edu/cs490f

Summary

- Project proposal due at start of lecture on Tuesday
- Next lecture on History of HCI
- Read
  - *As We May Think* by Vannevar Bush
  - *Tools For Thought* Ch 9 (Engelbart Demo)