(Designing) User Interfaces

G22.2280-001
Overview
Spring 05

Copyright Poelman & Associates, Inc.
(c) 2005. All rights reserved.
Me

• Professor Logan Poelman

• You can call me:
  – Logan
  – Mr. Poelman (pronounced like Pole – man)
  – Professor
  – Professor Poelman
Why are you here?

• Interested in understanding the art & science behind designing **usable** user interfaces
• Interested in the Web UIs or Java Swing UIs
• Want to learn something that you can apply in other courses, disciplines or vocations.
• Other?
What you can do …

• Please **attend** each and every lecture.
• Please be **prepared** for each lecture.
• Please get to lecture **on time**. That means be seated and ready to go by 7:00 PM
• To understand how to get the most out of this class see that section on the class web site.
• If you have **special needs** that can be accommodated to help you learn better in this course please let me know by sending me an email detailing your needs AND introduce yourself to me after class to let me know who your are.
• I’d like to learn the **names** of as many of you as I can. I’ll need your help to do that.
Caveats

• This course requires fluency in spoken and written English greater than most CS courses.
• This course is very open ended – you will not be told every step of every exercise
• You must:
  – Use your own judgment
  – Prioritize tasks
  – Take initiative
  – Work in teams and as individuals
  – Determine / create your own solutions
  – Invent, innovate and experiment
Course Description
(from the class site)

• Theory and practice GUIs
• Understanding the processes …
• Topics:
  – General design principles
  – General design process
  – Widget choice
  – Application design
  – Layout
  – Desktop UI
  – Web design
• Assignments … Java SWING based UIs … Web UIs …
• Discussion …
  – different UIs, different level users, Internationalization
  – GUI vs. non-graphical based UIs
  – Desktop, Web, Dedicated hardware, …
Course Description

• A course devoted to understanding the design of User Interfaces (UI).
• Primarily focused on Graphical User Interfaces (GUI).
• Design Process – a process to create a UI from idea to working prototype. Methods and techniques to facilitate the steps in the process.
• Design Tools – Paper, pens, whiteboards, wordprocessors and GUI/HTML editors.
• Design Components - understand the choices of layout, widgets, navigation, typefaces, images, colors, …
• Understand via the hands on design of several UIs.
This Course is Hard

- This is a **very hard course** for many students. I force you to think in ways that many students are not accustomed to. Many assignments are very open ended and "fuzzy." You will need to collaborate and create ideas with your group that you are assigned to. You will be asked to generate educated opinions about things.

- We focus on the theory **and practice** of graphical user interfaces (and other types of interfaces and interface technologies). We do this by creating imaginary future systems that will need UIs for them (think of me like the CEO of some crazy start up with a ridiculous ideas and a wad of cash ;-) )

- It is a course devoted to understanding the **processes** and tasks behind an iterative approach to designing good User Interfaces including:
  - idea generation
  - collaboration
  - gathering requirements
  - analysis
  - design
  - usability research
  - coding
  - web design
  - testing
  - alternative UI devices
  - refactoring

- Assignments will involve taking a problem **that no solutions actually exist for yet**, and creating the information that would lead you to being able to design a user interface for the solution to these problems.
Course Goals

• Learning to **think** about what constitutes a good UI and a bad UI
• Learning **techniques** to develop a good UI
• Understanding the **similarities** and **differences** between UI “platforms” i.e. desktop, web browser, PDA, kiosk, VRU, phone, dedicated console, …
• Difference between **Engineering** and **Design**
Not Course Goals

• Not to teach **graphic design/graphic arts**
• Not to teach you **Java** or **HTML**
• Not to teach a specific **platform**
• Not to concentrate only on **desktop** **GUIs,** **web** **GUIs** or even **GUIs** alone. Instead a bit of each and others.
## Course Schedule (see website)

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture Topic</th>
<th>Lecture Slides</th>
<th>Reading due date</th>
<th>Homework Assigned Date</th>
<th>Homework Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 24</td>
<td>1.) What is a UI?</td>
<td></td>
<td></td>
<td>HW#0</td>
<td></td>
</tr>
<tr>
<td>Jan 31</td>
<td>2.) A process - Design vs. implementation</td>
<td>EUID Chap 1 &amp; 2</td>
<td></td>
<td>HW#0</td>
<td></td>
</tr>
<tr>
<td>Feb 7</td>
<td>3.) UI Widgets &amp; Input Devices</td>
<td>EUID p.701-729 DWTW Ch.1</td>
<td></td>
<td>HW#1</td>
<td></td>
</tr>
<tr>
<td>Feb 14</td>
<td>4.) Swing Programming Intro &amp; Widgets</td>
<td>EUID p.53-107 &amp; p.337-516 DWTW Ch2,3,4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb 21</td>
<td>President’s Day - NO LECTURE</td>
<td></td>
<td></td>
<td>HW#2</td>
<td>HW#1</td>
</tr>
<tr>
<td>Feb 28</td>
<td>5.) What makes a good, bad or great UI? Why the Mac, Win, KDE all stink. Flow &amp; Csikszentmihalyi Layout &amp; Organizational structures</td>
<td>EUID p. 109-336</td>
<td></td>
<td>HW#2</td>
<td></td>
</tr>
<tr>
<td>Mar 7</td>
<td>6.) ACTIVITY IN CLASS: Collaborative Design Session - Please Attend.</td>
<td>DWSTW Ch 7, 10, 12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 14</td>
<td>SPRING BREAK - NO LECTURE</td>
<td></td>
<td></td>
<td>HW#3</td>
<td></td>
</tr>
<tr>
<td>Mar 21</td>
<td>7.) Application Design</td>
<td>EUID p. 517-700</td>
<td></td>
<td>HW#3</td>
<td></td>
</tr>
<tr>
<td>Mar 28</td>
<td>MIDTERM EXAM</td>
<td></td>
<td></td>
<td>HW#4</td>
<td>HW#3</td>
</tr>
<tr>
<td>April 4</td>
<td>8.) App Design</td>
<td>DWSTW Ch 5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>April 11</td>
<td>9.) App Design (cont)</td>
<td>DWSTW Ch 6</td>
<td>HW#5</td>
<td></td>
<td>HW#4</td>
</tr>
<tr>
<td>April 18</td>
<td>10.) Web Design - Cleaning up the WWW</td>
<td>DWSTW Ch 8,9</td>
<td></td>
<td>HW#4</td>
<td></td>
</tr>
<tr>
<td>April 25</td>
<td>11) Web Site Critiques</td>
<td>DMMT Complete Book</td>
<td></td>
<td></td>
<td>HW#5</td>
</tr>
<tr>
<td>May 2</td>
<td>12) Input &amp; Output</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May 9</td>
<td>FINAL EXAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Misc Class Info

• Office Hours
  – 6-7pm Mondays 4th Floor Warren Weaver Hall Room 401

• Class Web Page
  – check it every week
  – Suggest checking Monday before class
  – http://www.cs.nyu.edu/courses/spring05/G22.2280-001/index.html

• Class mailing list
  – http://www.cs.nyu.edu/mailman/listinfo/g22_2280_001_sp05
  – Subscribe with you OFFICIAL email address
  – Read email daily
  – Use it

• You needs a CIMS account or equivalent
  – email csgrad@cs.nyu.edu to request a new account.
TAs

• <<TBD>>
• TA office hours will be posted by second week of class on the class website.
Reading Assignments

• There are three **required** texts:

• **Optional** but recommended:

• You will be assigned reading assignments each week. You are expected to complete the readings **before** the next lecture.
• Readings will be part of the material for the **homework** and the **exams**.
• Several copies of the books will be place on reserve in the CIMS library.
• All should be available through www.amazon.com or the bookstore
• Amazon links:
Activities

• In class we will do a number of **activities** that students will be expected to participate in. The participation in the activities is voluntary but the lessons learned and content will be part of the material for the tests.

• Possible activities we will do in class include:
  – Focus Groups
  – Observations and Interviews
  – Collaborative Design Sessions (JAD)
  – Site Review & Critique – Play the game of “Trash the designer”
  – Developing Mockups
  – Developing 1st cut prototypes
  – Refining prototypes (SWING & Web GUIs)
Course Grades

- Homework – 30%
- Midterm – 30%
- Final – 40%
Grading Policy

• 1000 Total points
• 5 or 6 Homework Assignments (300 points total)
• 2 Exams (300 + 400 = 700 points)
  – Midterm (300 points)
    • 2 hours
    • Multiple choice, drawing and/or short answer
    • Reading, Lecture, Activities & Homework
  – Final (400 points)
    • 2 hours
    • Multiple choice and short answer
    • Reading, Lecture, Activities & Homework
    • Comprehensive (everything we’ve covered for the whole semester)
Grading Policy (cont)

<table>
<thead>
<tr>
<th>Homework</th>
<th>+ Midterm</th>
<th>Final</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Points</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Preliminary Grading Curve:
  - $\geq 900 \rightarrow A$
  - $\geq 800 \rightarrow B$
  - $\geq 700 \rightarrow C$
  - $\leq 650 \rightarrow F$
Homework

- Homeworks – still to be finalized. The content of which may include:
  - Field Observations
  - Gathering requirements, analyzing, designing and implementing designing
  - Task Analysis
  - Peer Project Reviews
  - Prototyping - Using paper, drawing programs, Java Swing IDE tools and HTML editing tools
  - Site reviews and critiques
  - Usability analysis
Project Tracks

• Cashier-less Checkout at the Supermarket
• Electronic Voting System
• Online Course Scheduling System
• Electronic Book
• Airport Reservation System – self checkin, online support, ordering – face recog based?
• Computer Intelligence Piloted Automobile Navigation System – a car that drives itself
• Online City Government System
• eWorkflow (like email on steroids)
• Home Personal Robot Servant
Homework Assignments

• 1 + 4 or 5 Homework Assignments
  – (0 + 5*60 = 300 points total)
  – Homework #0 is worth 0 points
  – Homeworks #1 – #5 are each 60 points
• Based on real world problems when possible.
• Submitted via email when appropriate.
• Will require you doing some field research.
• You will be required to **draw** using paper, pencil and/or drawing programs.
• You are **not** expected to be Artists
  but **neatness is important**.
• They will be posted on the day they are assigned. (I will send a class email to notify you when they are posted.)
Homework is due:

- Most will be submitted, via email to your TAs
- If to be submitted via EMAIL - Due before 5PM the day it is due. After that timestamp its considered late.
- If to be handed in, must be in my mailbox by 5PM or handed to me at the start of class (7:05 PM).
- If handed in AFTER I START CLASS , its considered late!
- Due dates will vary. Check the website.
- See the website for points deducted if late.
- I will discuss each assignment at the end of the lecture it is assigned in.
- Having problems understanding the homework assignment:
  - Email the class.
  - Contact your TAs.
  - Contact me.
Homework #0

• You need to download the NetBeans IDE
• You need to install it
• You need to:
  – create several simple Java Swing GUIs
  – zip them up
  – Email them to the TAs
• See the website for the SPECIFICS
• You need to do this by Jan 31st!
Resources

• Me
  – Via Office Hours, Email, Lecture

• The TAs
  – Via Office Hours, Email, Lecture
  – You will be assigned to a TA for grading purposes. You may contact either for help, though.

• The class website
  – http://cs.nyu.edu/courses/spring05/G22.2280-001
• Questions
• Comments

• 7.5
History of the GUI (briefly)
Definition of User Interface

• The mechanism by which a user interacts with a given system
• The mechanism that facilitates a user providing input to and/or receiving output from a system
• The point at which the combined user-system becomes the system and stops being the user. That surface that defines that boundary between the two.
User Interface

• The mechanism that supports a **dialog** between parties, a user and a system, usually.
• The mechanism supports, facilitates and enforces the syntax, grammar, conversational context, sequence of operations, and semantics & rules of the dialog between the parties in the dialog.
• The dialog is often a series of requests by one party for the other party to execute some action. These actions often result in state changes to the system. These changes may be publicly visible (or more accurately detectable as they may be non-visual) or private (non-visible). It includes changes in the state of the party or parties.
• Often this dialog is initiated and controlled by one party (user) more than the other (system) but this is not a requirement. A UI can support a peer to peer dialog where the user and/or the system can make requests.
User Interface

- **Dialog**
  - A conversation between two or more parties

- **Syntax**
  - Structure of statements made by

- **Grammar**
  - A set of Rules for constructing the statements

- **Context of the Dialog**
  - What is the implied subject? What is the current state of the system?

- **Sequence of Operations**
  - In what order are the statements allowed and if order will change the implied meaning of the statements
  - **Ex:** Open File, Write File, Read File vs. Open File, Read File, Write File

- **Semantics of the Dialog**
  - What is the meaning that the requesting party attaches to the statement, in the given context and the given order, and the expected reaction of the other speaker to the statement?

- **State**
  - Attributes of the system – color, shape, font, saved/unsaved, ....

- **Rules of the Dialog (a sort of FSM)**
  - What roles can each party play in the dialog?
  - What operations are permitted?
  - What operations are permitted in the current context which are not?
  - What will the result of a given operation be in the visible state indicators of the responding party?
History of user interfaces

• First tools created by humanoids
  – Stone tools with no handles
  – Stone tools with handles
• Hunting tools
  – Spears, knives,
• Shelter
• Domesticated Animals
• Fire
• Farming
Communication Interfaces

• Communication
  – Gestures
  – Verbal Symbolic – Noises, Grunts
  – Visual Symbolic – Ideograms
  – Verbal Language - 170+ languages/dialects currently spoken
  – Written Language – 40,000+ characters in UNICODE
    • http://www.unicode.org/charts/
Communication Interfaces

• Persistent Information Storage
  – Stone & Clay Tablets – Sumerian ??
  – Papyrus – Greek – 500 BCE
  – Paper – China – 100 ACE
  – Punched Cards – 1890’s
  – Magnetic Core Memory - 1940
  – Floppy disk – 1970
  – CDROM - 1980
Controls and Basic Latin

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>NUL</td>
<td>DLE</td>
<td>SP</td>
<td>0</td>
<td>@</td>
<td>P</td>
<td>`</td>
</tr>
<tr>
<td>1</td>
<td>SOH</td>
<td>DC1</td>
<td>!</td>
<td>1</td>
<td>A</td>
<td>Q</td>
<td>a</td>
</tr>
<tr>
<td>2</td>
<td>STX</td>
<td>DC2</td>
<td>&quot;</td>
<td>2</td>
<td>B</td>
<td>R</td>
<td>b</td>
</tr>
<tr>
<td>3</td>
<td>ETX</td>
<td>DC3</td>
<td>#</td>
<td>3</td>
<td>C</td>
<td>S</td>
<td>s</td>
</tr>
<tr>
<td>4</td>
<td>EOT</td>
<td>DC4</td>
<td>$</td>
<td>4</td>
<td>D</td>
<td>T</td>
<td>d</td>
</tr>
<tr>
<td>5</td>
<td>ENQ</td>
<td>NAK</td>
<td>%</td>
<td>5</td>
<td>E</td>
<td>U</td>
<td>u</td>
</tr>
<tr>
<td>6</td>
<td>ACK</td>
<td>SYN</td>
<td>&amp;</td>
<td>6</td>
<td>F</td>
<td>V</td>
<td>v</td>
</tr>
<tr>
<td>7</td>
<td>BEL</td>
<td>ETB</td>
<td>'</td>
<td>7</td>
<td>G</td>
<td>W</td>
<td>g</td>
</tr>
</tbody>
</table>

The Unicode Standard 4.0, Copyright © 1991–2003, Unicode, Inc. All rights reserved.

http://www.unicode.org/charts/PDF/U0000.pdf
Greek and Coptic

http://www.unicode.org/charts/PDF/U0370.pdf
Cyrillic

http://www.unicode.org/charts/PDF/U0400.pdf
Bengali, Arabic and Hebrew
What is being communicated by these two images? How?
Ideograms
Letterlike Symbols
Music Symbols
Braille
### Symbols

<table>
<thead>
<tr>
<th>260</th>
<th>261</th>
<th>262</th>
<th>263</th>
<th>264</th>
<th>265</th>
<th>266</th>
<th>267</th>
<th>268</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>☀️</td>
<td>☐️</td>
<td>☢️</td>
<td>☣️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
<tr>
<td>1</td>
<td>☐️</td>
<td>☐️</td>
<td>☒️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
<tr>
<td>2</td>
<td>☐️</td>
<td>☐️</td>
<td>☝️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
<tr>
<td>3</td>
<td>☐️</td>
<td>☐️</td>
<td>☔️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
<tr>
<td>4</td>
<td>☐️</td>
<td>☐️</td>
<td>☕️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
<tr>
<td>5</td>
<td>☐️</td>
<td>☐️</td>
<td>☔️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
<tr>
<td>6</td>
<td>☐️</td>
<td>☐️</td>
<td>☔️</td>
<td>☕️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
<td>☞️</td>
</tr>
</tbody>
</table>

- **7**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **8**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **9**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **A**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **B**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **C**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **D**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **E**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |

- **F**
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
  - ☐️  | ☐️  | ☐️  | ☐️  |
Computational Devices

• ??? - Abacus
• 1642 - Pascal’s adding machine
• 1670 - Liebnitz improves it (multiplication)
• 1800 - Jacquard Loom
  wooden boards that “program” the loom
• 1830 – Babage’s Difference Engine
  gears and levers
• 1890 - Hollerith Punched cards
UI Design Example

• Printing Press
• Movable Type - Gutenberg
• Typewriter
  – Invented by Mill – 1714
  – Remington 1873 (Sholes, Glidden & Soule’)
  – QWERTY – designed to limit speed of typing
  – Dvorak alternative layout
  – No real difference between good typist using QWERTY vs. Dvorak layouts
  – IBM Selectric – replaceable typeface balls
Modern Computers

• Punched Cards & Electronic Tabulating Machines – 1890 – IBM
• Howard Aiken – 1940’s
  – 1st digital
  – Punched paper tape
• Mauchley and Eckert – 1946
  – ENIAC – Electronic Numerical Integrator and Computer
  – Hardwired computer
Modern Computer UIs

• Punched cards – in/out
• Teletype, paper tape, print out
• CRT – command line interfaces
• Unix – 1969
  – Timesharing of the CPU
  – VI full screen character mode editor – Bill Joy of SUN fame – wrote it in a weekend
  – Emacs
  – X Windows (1980’s)
GUI

- **Sketchpad**
  - 1963
  - Sutherland at MIT
  - Light pen

- **Doug Englbart at Stanford Research Institute**
  - Started work in 1963
  - Windowed and used a mouse
  - Demoed 1968!

- [http://sloan.stanford.edu/mousesite/MouseSitePg1.html](http://sloan.stanford.edu/mousesite/MouseSitePg1.html)
Alan Kay

• Conceived of FLEX 1967
  – Tablet, windows, icons, hires
  – Never implemented
  – (created Smalltalk & Kaypro Computer)

• Dynabook 1969
  – Modelessness
  – Overlapping window
  – Unimplemented
    – www.artmuseum.net/w2vr/overture/integration.html
Athena and X Windows

- MIT in 1968 – Project Athena – to get standard interface to various workstations.
- X developed in mid 80’s
- Device independent windowing system
- 1988 MIT release X11R2 to the public
- http://www.x.org/
- KDE & Gnome are based on it.
Xerox PARC
(Palo Alto Research Center)

- Alto
  - 1972
  - Single User
  - Bitmapped Graphics
  - Mouse
  - WYSIWYG Editor – Bravo
  - 1 Meg removable HD
  - $7000 but never marketed
PARC

- Xerox STAR
- 1st commercial GUI based product
- $18000!
- Windows, Icons, Mouse Pointer (WIMP) UI
- Desktop and Office Metaphors
  - File Cabinet Icons
  - No direct manipulation (used menus)
- Unsuccessful in the market place
Apple in 1976 – Apple I
PC – Personal Computers

- MITS Altair 8800 – Toggle Switches, then keyboard, CRT and printer
- Apple II - 1977
- Commodore VIC / 64
- Sinclair / Timex ZX80
- TI99/4A
- IBM PC – 1981
  - 8 mHz, 320 Kbytes mem, dual floppy (no HD), 80 x 40 characters by 4 colors (CGA)
Apple visits PARC

- 1979
- Steve Jobs and the Lisa team
- Apple gives stock to Xerox in exchange for allowing them access
- Several PARC employees leave and go to Apple
Microsoft visits PARC

• Bill Gates and others 1980
Apple Lisa

• Released Jan 1983 ($12,000!)
• Multitasking, 1 meg ram!
• Direct manipulation (dragging with
• Hires B&W graphics (800 x 600)
• One Button Mouse
• WIMP (Windows, Icons, Mouse, Pointing) Interface
• Unreliable hardware / OS
• Failed in marketplace
Apple Macintosh

- January 1984
- Simpler version of Lisa
- 128K Ram, single tasking
- Desktop metaphor UI
- Built in UI toolkit in ROM
- B&W display
- No multitasking (initially)!
- First successful WIMP machine
- Originally didn’t have the file system metaphor for storage (more a database metaphor) but pressured into a file system metaphor.
Commodore Amiga 1000/512/2000

- WIMP
- Multitasking
- 256K memory or greater
- 640 x 480 x 4096 color!
- 4 channel wave based audio
- Special audio and video graphics chips – graphic accelerator chip!
- Not very successful in the business marketplace
- 1985!
MS Windows

- **V 1.0 1987**
  - Tiled windows (no overlaps)
  - No multitasking, max 1 meg ram
  - Unusable
- **V 2.0 - 1987**
  - Still weak features, 286 chip support!
- **V 3.x**
  - Finally usable
  - Multitasks, Overlapped Windows, Somewhat Stable
  - DTP (Desktop Publishing) – PageMaker – Killer App
- **“Windows 95” (V 4) - 1995**
  - Vast improvement, now close to a Mac for features
Next

- Steve Jobs left Apple in 1985
- 1988 1\textsuperscript{st} NextCube unveiled
- BSD 4.3 + Mach Kernal OS
- New window system not X
- Object Oriented UI (OOUI)
- 1000x1000 color resolution
- Rewritable Optical Drive (250 meg) removable NO HD! NO Floppy!
- Why is this relevant? (Hint: OS X)
PC GUI Timeline

1983 - Apple Lisa
1984 - Apple Mac
1985 - Amiga – multitasks, Color, Hi Res Video, Audio in 256K mem
1985 - Windows 1.x - Unusable – 1 meg address
1987 - Windows 2.x – Runs on a 286!
1988 - OS/2 – MS + IBM – Multitasks
1989 - System 7 for Mac
1989 - New Wave by HP
1989 - Steve Jobs leave apple forms Next
1990 - Windows 3.x - Finally somewhat usable
1992 - OS/2 Warp
1995 - Version 95
1993 - Win NT 3.1
1993 - NextStep 3.1 OS on Intel
1994 - Linux
1995 - Apple Newton released – Handheld Computer with handwriting recognition
1996 - Next is dead (or is it?)
1996 - Win NT 4.0
1998 - Win98
2002 - WinXP
Good sites for computing history

• The Smithsonian institution
  – http://www.si.edu/resource/faq/nmah/techhistory.htm

• Anthony J. Pennings, PhD - Marist College
  – http://www.academic.marist.edu/pennings/hyprhsty.htm

The Virtual Museum of Computing

http://vmoc.museophile.sbu.ac.uk/pioneers/
Different Computer Software
UI Devices

- Desktop GUI
- Web Browser
- PDA / Phone Browser
- Command line
- Phone
- Handwriting / TabletPC
- Speech?
- TV?
- Clothing?
- The Body?
- Others?
The Future

• Voice
• Tablet
• Writing
• Wearable Computers
  – Display
  – Waldos
• Avatars / Proxies
• 3D / 4D
• Virtual Reality
• RFID
• Thinking IF
• Questions
• Comments

• 8.0
What you need for this class

• A machine running Windows, linux + X or Mac
• NetBeans or some Java Swing IDE
• 500 mHz PIII or better
• 512 meg ram (256 is ok but may be slow)
• 500 meg disk space
• Color monitor + mouse
Homework #0

• Zero points but you still must do it
• Download NetBeans IDE (or some other SWING Designer IDE) and install it
  – Create 2 simple SWING based GUIs
• Zip up the GUIs and send them to your TA
• Download and install IE, Mozilla, Safari, and/or Opera Browsers (pick two or more for your platform)
• Subscribe to the class mailing list
Installing NetBeans

- [www.netbeans.org](www.netbeans.org)
- Latest Version ONLY!
- Supported on Windows, Linux and the Mac
- You can use other IDEs but the TAs and my self can only offer limited help with other editors.
- It MUST support Java 1.4 and Swing
- Ex:
  - JBuilder from borland
  - Dreamweaver by Macromedia
Using NetBeans

- Create a directory on your root that is called UIHomework
- Mount it in netbeans and/or create a project in netbeans
- Create a set of subdirectories: Homework0, Homework1, ..., Homework6
Create an Application
New >> Java GUI Forms >> Sample Forms >> Application

This template consists of a skeleton JFrame-based application, including three menus.
Call it “MyFirstSwingApp” in Homework0
Just click “next” until “Finish”
What you might see
To execute press the green triangle.
What you get if you run it.
The go back and create a JDialog called “MyFirstSwingDialog”
Then go to the layout and right click. Then select the absolute layout manager.
Using the Swing tab of widgets, place these into the dialog

• Add a JLabel
• Add a JButton
• Add a JTextBox
Should look like this
Change the title by editing the properties of the dialog. “My First Swing Dialog”
Executing it you should get …
Submitting your homework.

- The zip up your Homework0 directory and send it to your TA along with your full name and SID4 (last 4 of your SID) in the email.
- The TAs emails will be posted on the website.
- Due before lecture Jan 31 at 5PM!
NetBeans & Swing books

• Recommended:
  – “NetBeans: The Definitive Guide” by Tim Boudreau, Jesse Glick, Simeon Greene, Jack Woehr, Vaughn Spurlin
    – O’Reily - ISBN: 0596002807
Links

• Vannevar Bush Paper (Memex)
  – http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm link no longer works because of subscription requirements, how ironic ;-)  
  – Try here http://ccat.sas.upenn.edu/~jod/texts/vannevar.bush.html

• Ted Nelson (Xanadu and Hypertext)
  – http://xanadu.com/

• Who Is Jaron Lanier? (Virtual Reality Pioneer)
  – http://www.pcmag.com/print_article/0%2C3048%2Ca=22562%2C00.asp
  – http://www.well.com/user/jaron/