3. User Interface Design

Mircea F. Lungu

Based on a lecture by Oscar Nierstrasz.
Roadmap

- Interface design
- Design principles
- Graphical User Interfaces (GUI)
- Usability Testing
Roadmap

> **Interface design**
> **Design principles**
> **Graphical User Interfaces (GUI)**
> **Usability Testing**
The interface design process

User-Interface (UI) design is an *iterative process* involving close liaisons between users and designers.

The 3 core activities in this process are:

— *User analysis*. Understand what the users will do with the system;
— *System prototyping*. Develop a series of prototypes for experiment;
— *Interface evaluation*. Experiment with these prototypes with users.
The design process

- Analyse and understand user activities
- Produce paper-based design prototype
- Evaluate design with end-users
  - Design prototype
- Produce dynamic design prototype
  - Evaluate design with end-users
  - Executable prototype
- Implement final user interface

© Ian Sommerville 2004
Personas

It is sometimes better not to talk about “the user” but think about a clear customer.

**Technique from Marketing**
- generated after interviews with users
- helps in focusing a product’s features
- a single persona should be the main focus a design

Popularized by Alan Cooper in his book “The Inmates are Running the Asylum”
Roadmap

- Interface design models
- **Design principles**
- Graphical User Interfaces (GUI)
- Usability Testing
User Interface Design Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User familiarity</td>
<td>Use terms and concepts <em>familiar</em> to the user.</td>
</tr>
<tr>
<td>Consistency</td>
<td>Comparable operations should be activated in the <em>same way</em>. Commands and menus should have the same format, etc.</td>
</tr>
<tr>
<td>Minimal surprise</td>
<td>If a command operates in a known way, the user should be able to <em>predict</em> the operation of comparable commands.</td>
</tr>
<tr>
<td>Feedback</td>
<td>Provide the user with visual and auditory feedback, maintaining <em>two-way communication</em>.</td>
</tr>
</tbody>
</table>
## User Interface Design Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Memory load</strong></td>
<td>Reduce the amount of information that must be remembered between actions. <em>Minimize</em> the memory load.</td>
</tr>
<tr>
<td><strong>Efficiency</strong></td>
<td>Seek efficiency in dialogue, motion and thought. <em>Minimize keystrokes and mouse movements.</em></td>
</tr>
<tr>
<td><strong>Recoverability</strong></td>
<td>Allow users to <em>recover from their errors</em>. Include undo facilities, confirmation of destructive actions, 'soft' deletes, etc.</td>
</tr>
<tr>
<td><strong>User guidance</strong></td>
<td>Incorporate some form of <em>context-sensitive user guidance</em> and assistance.</td>
</tr>
</tbody>
</table>
Roadmap

- Interface design models
- Design principles
- Graphical User Interfaces (GUI)
- Usability Testing
With a command language, the user types commands to give instructions to the system

- May be implemented using cheap terminals
- Easy to process using compiler techniques
- Commands of arbitrary complexity can be created by command combination
- Concise interfaces requiring minimal typing can be created
Command Interfaces

**Advantages**
> Allow experienced users to *interact quickly* with the system
> Commands can be *scripted* (!)

**Problems**
> Users have to *learn and remember* a command language
> Not suitable for *occasional* or inexperienced users
> An *error detection* and recovery system is required
> *Typing ability* is required (!)
XEROX Alto was the first computer to use the desktop metaphor. And a mouse.

More about the history of XEROX PARK in Dealers of Lighting.
GUIs

**Advantages**

> They are *easy to learn* and use.
> Users without experience can learn to use the system quickly.

> The user may *switch attention* between tasks and applications.

> *Fast, full-screen interaction* is possible with immediate access to the entire screen

**Problems**

> A GUI is not automatically a good interface
> Many software systems are *never used* due to poor UI design
> A poorly designed UI can cause a user to make *catastrophic errors*
## Components

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Windows</strong></td>
<td>Multiple windows allow <em>different information to be displayed simultaneously</em> on the user’s screen.</td>
</tr>
<tr>
<td><strong>Icons</strong></td>
<td>Usually icons represent <em>files</em> (including folders and applications), but they may also stand for <em>processes</em> (e.g., printer drivers).</td>
</tr>
<tr>
<td><strong>Menus</strong></td>
<td>Menus bundle and organize <em>commands</em> (eliminating the need for a command language).</td>
</tr>
<tr>
<td><strong>Pointing</strong></td>
<td>A pointing device such as a mouse is used for <em>command choices</em> from a menu or indicating items of interest in a window.</td>
</tr>
<tr>
<td><strong>Graphics</strong></td>
<td>Graphical elements can be <em>commands</em> on the same display.</td>
</tr>
</tbody>
</table>

Supporting Consistency and Minimal Surprise.
Menu Systems

Advantages

> Users don’t need to remember command names
> Typing effort is minimal
> User errors are trapped by the interface
> Context-dependent help can be provided (based on the current menu selection)

Problems

> Actions involving logical conjunction (and) or disjunction (or) are awkward to represent
> If there are many choices, some menu structuring facility must be used
> Experienced users find menus slower than command language
Menu Structuring

**Scrolling menus**
> The menu can be scrolled to reveal additional choices
> Not practical if there is a very large number of choices

**Hierarchical menus**
> Selecting a menu item causes the menu to be replaced by a sub-menu

**Walking menus**
> A menu selection causes another menu to be revealed

**Associated control panels**
> When a menu item is selected, a control panel pops-up with further options
Colour Use Guidelines

Colour can help the user understand complex information structures.

> Don’t use (only) colour to communicate meaning!
  — Open to misinterpretation (colour-blindness, cultural differences ...)
  — Design for monochrome then add colour

> Use colour coding to support user tasks
  — highlight exceptional events
  — allow users to control colour coding

> Use colour change to show status change

> Don't use too many colours
  — Avoid colour pairings which clash

> Use colour coding consistently
Platform Specific GUI Patterns

http://developer.android.com/design/patterns
Roadmap

- Interface design models
- Design principles
- Graphical User Interfaces (GUI)
- Usability Testing
Observe a group of test subjects performing a pre-defined scenario

— Which test subjects?
— How many test subjects?
— Which scenarios?
— What to observe?

Jakob Nielsen, *Usability Engineering*
Some evaluation of a user interface design should be carried out to assess its *usability*.

Full scale evaluation is very *expensive* and *impractical* for most systems.

Ideally, an interface should be evaluated against a *usability specification*. However, it is rare for such specifications to be produced.
Simple evaluation techniques

> **Questionnaires** for user feedback.
> **Video recording** of system use and subsequent tape evaluation.
> **Instrumentation** of code to collect information about facility use and user errors.
> The provision of code in the software to collect *on-line user feedback.*
Hints

> Establish concrete goals — what do you want to achieve?
  — What criteria will you use to establish “success”?
  — What data will you collect?
  — Choose representative test tasks.

> Carry out a pilot test first.

> Test users should truly represent the intended users.

> Use experienced experimenters. (Get trained!)
  — Make the test subjects feel comfortable.
  — Don’t bias the results.
### Usability Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Learnability</strong></td>
<td>How long does it take a new user to become <em>productive</em> with the system?</td>
</tr>
<tr>
<td><strong>Speed of operation</strong></td>
<td>How well does the system <em>response</em> match the user’s work <em>practice</em>?</td>
</tr>
<tr>
<td><strong>Robustness</strong></td>
<td>How <em>tolerant</em> is the system of user error?</td>
</tr>
<tr>
<td><strong>Recoverability</strong></td>
<td>How good is the system at <em>recovering</em> from user errors?</td>
</tr>
<tr>
<td><strong>Adaptability</strong></td>
<td>How closely is the system tied to a <em>single model</em> of work?</td>
</tr>
</tbody>
</table>
Why you need to test with 5 users


http://www.useit.com/alertbox/20000319.html
Roadmap

> Interface design models
> Design principles
> Graphical User Interfaces (GUI)
> Usability Testing
> **Summary**
Key points

> The user interface design process involves *user analysis, system prototyping* and *prototype evaluation*.

> *User interface design principles* should help guide the design of user interfaces.

> *Interaction styles* include direct manipulation, menu systems form fill-in, command languages and natural language.

> *Graphical displays* should be used to present trends and approximate values. *Digital displays* when precision is required.

> *Colour* should be used *sparingly and consistently*.

> The goals of *UI evaluation* are to *obtain feedback* on how to improve the interface design and to assess if the interface meets its *usability requirements*. 
What you should know!

- Interface design principles
- What are personas and why are they useful
- Trade-offs between menus and command languages
- How to use color to improve a UI
- Android UI design patterns
Can you answer the following questions?

> Why is it important to offer “keyboard shortcuts” for equivalent mouse actions?
> How would you present the current load on the system? Over time?
> What is the worst UI you ever used? Which design principles did it violate?
> What’s the worst web site you’ve used recently? How would you fix it?
Sources

Recommended reading
> Alan Cooper, *The Inmates are running the Asylum*, SAMS, 1999.
> *The Interface Hall of Shame*, (link)

http://homepage.mac.com/bradster/iarchitect/shame.htm
http://www.frankmahler.de/mshame/
Attribution-ShareAlike 3.0

You are free:
- to copy, distribute, display, and perform the work
- to make derivative works
- to make commercial use of the work

Under the following conditions:

**BY:** Attribution. You must attribute the work in the manner specified by the author or licensor.

**Share Alike.** If you alter, transform, or build upon this work, you may distribute the resulting work only under a license identical to this one.
- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

http://creativecommons.org/licenses/by-sa/3.0/