GUI Bloopers

Basic Principles

What is a GUI Blooper?

• “Bloopers” are mistakes that software developers frequently make when designing graphical user interfaces
  – Not just specific examples but mistakes that developers make over and over
  – Goal is to give examples of mistakes AND help designers and developers learn to produce better GUIs
Basic Principle 1

• Focus on the users and their tasks, not on the technology

• Requires collaboration with the user

• “Software should be designed neither for users nor by them, but rather with them.”

Basic Principle 2

• Consider function first, presentation later

• We’ve said that you should make a GUI mock-up before coding, isn’t this contradictory?
  – No, function means determining the requirements and basic functions of the software. It doesn’t mean writing actual code functions.
  – Before GUI layout we must decide what data the users can create, view, or manipulate.

  • Conceptual Model
Basic Principle 3

• **Conform to the user’s view of the task**

• Software user interfaces should be designed from the user’s point of view
  – Obviously this requires that you know what the user’s point of view is (Basic Principle 1)
  – Strive for naturalness
    • E.g. in chess, drag and drop piece or enter coordinates?
  – Don’t impose arbitrary restrictions
    • E.g. maximum of 255 entries
  – Use user’s vocabulary, not your own
  – Keep program internals inside the program
    • Includes error messages

Basic Principle 4

• **Design for the common case**

• Ever create a new “object” in a program and find yourself having to change its default properties all the time?
• Strive to make common tasks easy
• Sensible defaults, templates or “canned” solutions, wizards, customizability
Basic Principle 5

• **Don’t distract users from their goals**

• People are good at multi-tasking, but not for problem solving and stuff we don’t do all the time. Software shouldn’t distract users from their own tasks and goals.
  
  – E.g. hard to find functions, confusing terminology

• Operate in the background, not the foreground of user’s consciousness.

Basic Principle 6

• **Facilitate Learning**

• Software is often hard to learn, some of the blame may be from “inside-out” thinking, the idea that users will actually know how to operate the software
  
  – Clear to developers but may not be clear to users
Basic Principle 6

• Example: Graphical Ambiguity
  – With lots of icons it is difficult to make them meaningful
  – What does this mean?

Antenna for a transmit function, not a martini glass with a stick

Basic Principle 7

• Deliver information, not just data

• E.g. comparative chart vs. list of data values
• Don’t treat the data like information, focus on the important data and extract necessary information from it
Basic Principle 7

- The screen belongs to the user
  - Don’t take over the screen, researchers discovered it is usually a bad idea for software to unilaterally move controls and data around on the screen
    - Jump or “warp” the mouse to new positions
    - Move something to the mouse location
    - Reposition windows
    - Automatically rearrange data for the user
  - Controlling the mouse for the user violates the hand-eye coordination a user has with the machine

Basic Principle 7

- Preserve “Display Inertia”
- When software changes a display to show the effect of a user’s actions, it should try to minimize what it changes
  - Small local changes should produce small, local changes on the display
  - Attempt to keep as much of the display unchanged as possible
- Helps the user retain context, minimizes disruption
- Examples of poor display inertia:
  - Forcing entire page to refresh
  - Scrolling to a different position in the browser
Basic Principle 8

• **Design for responsiveness**

  • A software application’s ability to keep up with users and not make them wait
    – The most important factor in determining user satisfaction
    – Users hate waiting more than anything else
  
• **Desire for speed is perceived, not actual**

  – A responsive interface that shows progress in computing a result is perceived as faster than one that displays nothing (system pauses) until the result is done

Basic Principle 8

• Systems can be slow in terms of performance but still responsive

  – Might queue requests but never lock-up or force users to wait for the system to catch up

• Examples of poor responsiveness

  – Delayed feedback for button-press or mouse click
  – Operations that block activity
  – No visual feedback as to how long a lengthy operation will take
  – Jerky animation

![Effect of response time on user productivity](image)
Real Time Interface

- 0.1 seconds
  - Limit for perception of cause-and-effect between events
  - Software that waits longer than 0.1 seconds to register a reaction to a user action appears "broken"
  - Limit for perception of smooth animation
- 1 second
  - Maximum comfortable gap in a conversation
  - If displaying information on the screen the user is unlikely to react until at least one second
- 10 seconds
  - Unit of time into which people break down their planning and execution of larger tasks
  - Every ten seconds user like to look up and reassess their task status, relax, etc.
  - Like to mark a task complete and move onto the next one
  - Amount of time a user is willing to spend to set up and operation and start it before losing patience (operation can take longer)

Basic Principle 9

- Try it out on users, then fix it!
- Test early and often, results may surprise even experienced designers
- Schedule time to correct problems found by tests
- Tests have two goals
  - Information on aspects of the UI that cause difficulty
  - Socially it convinces developers that there are design problems that need correcting. Some developers need to see users have problems for themselves.
GUI Control Bloopers

• Two categories of control bloopers
  – Using the wrong GUI Control
  – Using a control incorrectly

• Control bloopers harm usability and give customers an impression of a shoddy, unprofessional product

Blooper 1: Confusing checkboxes and radio buttons

• Radio buttons when only one is selectable
• Checkboxes when many selectable

Word: Change font, effects, subscript and superscript

Color: Red
**Blooper 1**

- Diebold/Premier AccuVote TSx operates somewhere between checkboxes and radio buttons

**Avoiding Blooper 1**

- **Use radio buttons**
  - When only one option may be selected
  - In sets of at least two
  - Ensure enough space is available to see all options
  - The number of options is fixed and small (2-8)
- **Consider dropdown or scrolling menus which requires less space**
- **Checkboxes represent ON/OFF conditions that are independent of each other**
Blooper 2: Checkboxes for non-ON/OFF Setting

- Checkboxes should be used for on/off not for a selection of items.

  Sort Order: □ Ascending

- Instead use radio buttons.

  Toolbar: ● Horizontal ○ Vertical

Blooper 4: Using tabs as radio buttons

- Misuse of tabs is to use them as if they are radio buttons to present choices that affect what the application will do rather than just which controls are displayed.

- Some users will not realize the last tab selected is the one that is used – users expect tabs just for switching between panels.
Tabs as Radio Buttons

• Better design:

![Example of tabs]

Tabs should be purely navigational controls, not for settings

Blooper 5: Too Many Tabs

• Intended to save space but too many uses more space – usually doesn’t scale beyond a handful

![Example of blooper 5]

• Never use dancing tabs; change position based upon which tab is selected
  – Unavoidable with multi-rows of tabs
Multi-Row Tabs

Solutions: Widen panel, make tabs narrower, or use another control instead of tabs

Blooper 6: Using input controls for display-only data

• Don’t use input controls (textboxes, radio buttons, checkboxes, etc.) to present data users cannot change. This refers to controls that are never editable, not to ones that are temporarily inactive (grayed out).
Blooper 6
Example

Better uBid Page
Blooper 6 Example

National Geographic Trip Planner: trip Origin and Destination fields look like directly editable text, but are only indirectly editable via “Select a City...” buttons and dialog boxes.

Avoiding Blooper 6: Use labels, don’t use controls that look like they can be edited.

Blooper 7: Overusing text fields for constrained input

- Text fields are too unstructured for constrained data
  - Dates, postal codes, volume levels, monetary amounts, etc.
  - Especially occurs in paper to GUI conversion
- Use structured controls to allow only valid data
Take advantage of the GUI

Instead of:

<table>
<thead>
<tr>
<th>Birthdate: 4/21/52</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telephone Number: (415) 555-1212</td>
</tr>
</tbody>
</table>

Use:

<table>
<thead>
<tr>
<th>Birthdate: 4/21/52</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telephone Number: (415) 555-1212</td>
</tr>
</tbody>
</table>

Instead of:

| Appt. Time: 11:30 am |

Use:

| Appt. Time: 11:30 am or |

| Appt. Time: 11:30 pm |

Database Project w/Job Titles

- Contractor is trying to match up titles, but getting values like:
  
  Consultant(Nonexempt)
  Consultant(Nonexempt)
  Consultant (Noneexempt)
  Consultant (Nonexempt)
  Consultant (Nonexempt)
  Consultant(Nonexempt)
Blooper 9 : Intolerant Data Fields

• To be friendly and helpful your text fields should tolerate reasonable variations in what people type
  – E.g. filter out spaces (common w/cut and paste), dashes, period, tab, etc.

![Payment Options](image1)

Stuffit.com: rejects credit card numbers with spaces.

Blooper 9 Example

![Mileage Plus login](image2)

http://www.united.com

United.com: rejects Frequent Filler numbers with spaces, the format United uses elsewhere.
Avoiding Blooper 9

• Match field length to data
  – Visible length suggests how much to type
• Accept common formats
• Beware of rejecting legitimate data
• Make case irrelevant
• Provide a pattern (e.g. draw dashes in QP-00-3412)
• Structure text fields
  – Use pull-down menus or combo-boxes

Blooper 10: Input fields and controls with no default

• Defaults should be set up with the most likely values; users only need to scan the settings, change a few, and proceed
Blooper 10 Examples

• Drop-down menu with no default and poor labeling

StanfordBookstore.com: dropdown menu with no default value and poor labeling.
Avoiding Blooper 10

• Use likely default values
• Always add initial value for radio buttons, or “None” as an explicit choice

Blooper 11: Poor Defaults

• A default value that is unlikely to be what users want is more harmful than no default value
Blooper 12: Negative Checkboxes

- Negative checkboxes turn a feature or attribute OFF when checked and ON when unchecked

![SmartDraw: negative checkbox — checking it turns spell-checking OFF.](image1)

![ExactTarget.com: user-permission settings include checkboxes that “remove” permissions.](image2)

Blooper 12 Example

- SQL Server Enterprise Manager
  - Deny or Allow access

- Avoiding Blooper 12
  - All checkboxes should be positive
Navigation

• The most pervasive problem software users encounter is navigation: finding their way to what they are seeking.

• People should know
  – Where they are
  – Where they’ve been
  – Where they can go
  – Whether the goal is near or far

Blooper 13: Window or page not identified

• Some applications or websites fail to provide any sign of where the user is. One failure is to provide a window title:
Avoiding Blooper 13

- Title all windows, including dialog boxes. Use the format:
  
  `<AppName>:<WindowTitle>`

Blooper 14: Same title on different windows

- Sometimes different windows or web pages have the exact same title; this can mislead users about where they are.
Unique Window Names

Blooper 15: Window title doesn’t match command or link

- Users need reassurance that they got what they wanted to get. Avoid haphazard mappings between commands or links and the windows or pages they display.
- Mismatched titles can mislead users into thinking they selected the wrong command.
Blooper 15 Example

Microsoft Excel: Insert => Function... command displays Paste Function dialog box.

Blooper 15 on the Web

mac.com WebMail: clicking "Add a folder" displays a window titled "Manage Folders."
Related Problem: Mystery Meat Navigation

- This is when you have to click on something to figure out what it is.
- Examples:
  - http://www.flatpakhouse.com/
  - http://www.daltonmailingservice.com
  - http://www.shmarketing.co.uk/

Avoiding Blooper 15

- Make titles of windows or web pages match the command that displays them
- Inexact matches are OK if they work for users
  - As long as users see the connection
  - E.g. “Show Order Status” and “Status of Order #52” would be minor enough not to confuse anyone
Blooper 16: Distracting off-path buttons and links

- People follow an “information scent” to their goals based on cues in the interface. Software should help provide proper “scents” to guide users and not lead them astray.

Blooper 16 Example

- Too many distracting links for IEEE renewal page

Renew Membership

We've made changes to the online renewal experience for 2008. Review this quick reference guide for step by step instructions.

Begin Renewal Now >>

Instructions for IEEE Student Members
For 2008 all student members must renew or add new services online (join IEEE societies or add publication subscriptions). For more information, refer to the IEEE Student Concourse.

For all IEEE Members
You will need your IEEE Web Account to access the secure online renewal application.
Accepted online payment methods: MasterCard or Visa, MasterCard, Visa, American Express, Diners Club

- Consider adding the IEEE Member Digital Library to your membership when you renew for 2008.
- Download or print PDF versions of the 2008 IEEE Special Interest Memberships & Subscriptions catalog, for detailed descriptions and pricing information for all offerings. (Requires Adobe Acrobat Reader®; available from the Adobe web site.)
- Read important information about IEEE and 3rd Party Mailings, as well as the IEEE Subscription Delivery Policy.

Questions? View our Frequently Asked Questions or Contact IEEE Member Services.
Blooper 16 : Confusing Links

Users may skip to “continue” and click it instead of “Submit Information”
Avoiding Blooper 16

• Don’t distract the customer from their task, let them finish the primary task first
• Create a “process funnel” that guides users toward their goal
• Make sure off-path button or link labels don’t trick users into thinking they are on the same path.
  – Could use pop-up or tooltip windows to show explanations

Blooper 17: Self-links

• Some web pages have links to themselves
  – Disorienting as users may not recognize the redisplayed page as the one they are on
Avoiding Blooper 17

• Don’t include active links to the current page on the current page
• Don’t forget about the navigation bar
  – Don’t include a link to the current page
  – Helps illustrate the current page as well
  – In breadcrumb, avoid link to “here”

![Breadcrumb Example](image)

United.com: last item (for current page) in breadcrumb path is not a link.

Blooper 18: Diabolical Dialog Boxes

• Too Many Levels
  – Deep hierarchies divert users from original goals, lose track of which OK, Apply, Cancel buttons are before them
  – Most people lose track more than a few levels down a hierarchy

![Dialog Box Example](image)

Apple FinalCut Pro: exporting image from video, optimized for streaming, requires (A), (B), (C), (D), and (E).
Avoiding Blooper 18

- General rule: avoid more than two levels of dialog boxes
  - Rule only applies to dialog boxes
  - Some dialog boxes don’t count
    - E.g. a File chooser or error messages
- Ways to cut excess levels
  - Chart the window hierarchy
  - Use a “details” pane instead of a separate window

Blooper 20: Poor search results browsing

- Users should be able to browse search results efficiently
- IBM search result for “tablet computers”
  - Only “Back” and “Next” buttons for 1438 results
Blooper 20 Example

• Search from [www.buyreliant.com](http://www.buyreliant.com) has only Back/Next with 20 hits per page and no Totals

Avoiding Blooper 20

• Search results should remind users what the search terms were, indicate number of hits, and make it easy to browse through the hits.
Textual Bloopers

Blooper 23 : Unclear terminology

• Ambiguous terms, e.g. “enter”
• Terms for different concepts overlap in meaning
• Concepts too similar

Eudora for Mac: users must learn arbitrary distinction between Find and Search.
Blooper 24 : Bad writing

• Makes software look amateurish even if it doesn’t hurt usability
• Inconsistent writing style
  – Terse in some places, verbose in others
  – Commands named after verbs in some places and named after nouns in others
  – Different capitalization
  – Ending some but not all sentences with periods

Blooper 24

• Variation: Poor diction, grammar, spelling, punctuation
Blooper 25: Too much text

• Needless text is bad anytime, but especially in software when it can distract users from their goals

Blooper 25 Example

• Wordy and repetitious text from dmv.ca.gov

Driver License Information for New Drivers

Driver License Information for Persons Over 18

- How to apply for a driver license if you are over 18
- How to apply for a commercial driver license (CDL)
- How to apply for a motorcycle or moped driver license if you are over 18

Provisional Driver Permit and License Information for Persons Under 18

- How to apply for a provisional permit if you are under 18
- Parent or guardian signatures - Accepting liability for a minor
- Driver Education and Driver Training Information
- Provisional driver license restrictions during the first year
- Teen Driver Information
- How to apply for a motorcycle or moped driver license if you are under 18

Application Requirements and Acceptable Forms of Payment

- What documents can I use to verify my birth date and legal presence?
- Do I need my social security number (SSN)?
- How does DMV check my record?
- What are the alternative methods for completing the driver's permit written examination?
- What other languages is the written or audio test available in?
- Who can sign for my permit or license if I'm under 18?
- What show it mean when they sign for me?
- What medical report do I need for a Commercial License?
- What methods of payment are acceptable by DMV services?
Example cutting needless text

Jeep.com website from early 2002 to late 2002 to 2007

Blooper 26: Geek Speak

• Easy to allow programmer jargon to seep into the end product (assuming non-tech end user)
  – Error while checking mail
    TCP/IP Error 706; {37:1253}
  – Interface Hall of Shame:
    • A caller to Compuserve customer support said that even though he did what the software told him to do, it didn’t work.
Blooper 27 : Calling users “user” to their face

• “Users” is what software developers call people who use our systems. It’s a fine term when talking to other developers, but it is not what users call themselves.
• Only two industries call their customers “users.” One is computer software. Do you know what the other industry is?

Blooper 27 Examples
Blooper 27 Examples

Avoiding Blooper 27

- Use a non-developer term like “visitor”, “customer”, or “member” instead of “user”.
Blooper 28 : Vague error messages

• Related to geek-speak are error messages that give vague, generic errors instead of being helpful to the user.

• Variations
  – Messages displayed by low-level code
  – Reason for error not given to higher level code
  – Generic message components

Blooper 28 Examples

• Some examples:
  – “Nesting level too dip.”
    • Burned into ROM and shipped tens of thousands
  – “Error 500 HTTP Web Server”
  – “Excuse me, but Eudora could use some help.”
  – “File missing or you don’t have access.”
  – “Name contains invalid characters.”
  – “Value of field exceeds limit.”
The Winner of Vague Error Messages

Avoiding Bloopers 28

- Express the error in terms of the task
- Don’t just identify the problem; suggest a solution
  - Messages should contain:
    - Error symbol; Problem: Solution
- Pass errors up to code that can translate them for users.
- Design messages and message-bearing components to accept details at runtime
  - As opposed to a static error message with no runtime details
- Different types of messages have different audiences
  - User errors: end users
  - Logs: system admin
  - Debugging/tracing: developers