

EKSISTERENDE HCI-RETNINGSLINJER FOR BRUKERGRENSESNITT PÅ PC / WEB

Oktober 2006

Redaktør: Riitta Hellman

Karde AS Tellu AS



Dette notatet oppsummerer *eksisterende* retningslinjer for utforming av brukerdialogen på PC-plattform. Ofte – men dog ikke alltid – er slike retningslinjer ment å dekke webgrensesnitt. Notatet er en delleveranse fra det IT Funk-finansierte prosjektet 'Norsk OSIRIS og universell utforming'. I prosjektet skal disse retningslinjene brukes sammen med retningslinjer som er rettet mot digital/interaktiv TV og mobiltelefon (jf. egne notater).

Nedenfor presenteres et lite antall eksisterende retningslinjer som til sammen dekker feltet svært godt. Til en viss grad viser de også til hverandre. Vi har lagt vekt på kjente og anerkjente retningslinjer. Det er ikke utarbeidet oversettelse til norsk i og med at disse kun bli brukt som referansemateriale.

Vi har også tatt med prinsippene for universell utforming i deres opprinnelige form. Disse prinsippene er styrende for prosjektets målsettinger om universell utforming i mobiltelefonen og den interaktive TVns brukergrensesnitt mot tjenester.

1. Jacob Nielsen: Ten Usability Heuristics

Kilde: http://www.useit.com/papers/heuristic/heuristic_list.html

1. VISIBILITY OF SYSTEM STATUS

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

2. MATCH BETWEEN SYSTEM AND THE REAL WORLD

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

3. USER CONTROL AND FREEDOM

Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

4. CONSISTENCY AND STANDARDS

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

5. ERROR PREVENTION

Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

6. RECOGNITION RATHER THAN RECALL

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

7. FLEXIBILITY AND EFFICIENCY OF USE

Accelerators – unseen by the novice user – may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

8. AESTHETIC AND MINIMALIST DESIGN

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

9. HELP USERS RECOGNIZE, DIAGNOSE, AND RECOVER FROM ERRORS

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

10. HELP AND DOCUMENTATION

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

2. Ben Shneiderman: Eight Golden Rules

Kilde: http://wps.aw.com/aw_shneider_dtui_4/0,8896,1257130-,00.html

1. STRIVE FOR CONSISTENCY

Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent commands should be employed throughout.

2. ENABLE FREQUENT USERS TO USE SHORTCUTS

As the frequency of use increases, so do the user's desires to reduce the number of interactions and to increase the pace of interaction. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.

3. OFFER INFORMATIVE FEEDBACK

For every operator action, there should be some system feedback. For frequent and minor actions, the response can be modest, while for infrequent and major actions, the response should be more substantial.

4. DESIGN DIALOG TO YIELD CLOSURE

Sequences of actions should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment.

5. OFFER SIMPLE ERROR HANDLING

As much as possible, design the system so the user cannot make a serious error. If an error is made, the system should be able to detect the error and offer simple, comprehensible mechanisms for handling the error.

6. PERMIT EASY REVERSAL OF ACTIONS

This feature relieves anxiety, since the user knows that errors can be undone; it thus encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.

7. SUPPORT INTERNAL LOCUS OF CONTROL

Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.

8. REDUCE SHORT-TERM MEMORY LOAD

The limitation of human information processing in short-term memory requires that displays be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.

3. The GNOME Usability Project: GNOME Human Interface Guidelines 2.0

Kilde: <http://developer.gnome.org/projects/gup/hig/>

1. DESIGN FOR PEOPLE

Remember that the purpose of any software application is to enable some group of people to accomplish a specific set of tasks. So, the first things to establish when designing your application are:

- who your users are and
- what you want to enable them to do.

2. DON'T LIMIT YOUR USER BASE

If you are designing an application for use by engineers, or by children, or by system administrators, be sure to create an application that can be used by *all* engineers, children, or system administrators, including those with disabilities or those who are native speakers of a language different from yours. Be aware of accessibility issues and internationalisation and localization issues. Accessibility means enabling people with disabilities of some kind to participate in life's activities: in this case, specifically to use your software. Internationalisation means designing software so that it can function in different language environments. Localisation is the process of actually translating the messages, labels, and other interface elements of an application into another language.

3. CREATE A MATCH BETWEEN YOUR APPLICATION AND THE REAL WORLD

Always use words, phrases, and concepts that are familiar to the user rather than terms from the underlying system. Use terms that relate to the user's knowledge of the tasks your application supports.

4. MAKE YOUR APPLICATION CONSISTENT

Make your application consistent with itself and with other applications, in both its appearance and its behaviour. This is one of the most important design principles, and probably the most famous, but it is also frequently ignored.

5. KEEP THE USER INFORMED

Always let the user know what is happening in your application by using appropriate feedback at an appropriate time. The user should never have to guess about the status of the system or of your application. When the user performs an action, provide feedback to indicate that the system has received the input and is operating on it. Feedback can be visual, audio, or both. If the system will take a long time to process the request, provide as much feedback as possible about how lengthy the operation will be. Types of helpful feedback include but are not limited to: cursor changes, animated "throbbers", progress indicators, audio feedback such as a beep, and error messages. Error messages should use simple language, clearly state the problem, and provide solutions or tell the user how to get out of the current situation if possible.

6. KEEP IT SIMPLE AND PRETTY

Your application should enable the user to concentrate on the task at hand. So, design your application to show only useful and relevant information and interface elements. Every extra piece of information or interface control competes with the truly relevant bits of information and distracts the user from important information. Hence, don't clutter your interface, and don't overload the user with buttons, menu options, icons, or irrelevant information. Instead, use progressive disclosure and other techniques to limit what the user sees at any given moment. Present your information and interface elements in an aesthetically pleasing manner.

7. PUT THE USER IN CONTROL

Remember that computers exist to serve humans. A user should always feel in control, able to do what they want when they want. This means you should generally avoid modes; users should be able to switch between different tasks (and specifically, different windows) at any time. The user should also be able to tailor aspects of their environment

to fit personal preferences. It is very important, however, to avoid the trap of allowing too much configuration, or allowing the configuration of parameters that most users will not understand or find useful to modify. Wherever possible, inherit visual and behavioural parameters from global preferences and settings.

8. FORGIVE THE USER

Your application should therefore allow users to quickly undo the results of their actions. If an action is very dangerous, and there is no way to undo the result, warn the user and ask for confirmation. Only do this in extreme cases, though; if frequently faced with such confirmation messages, users begin to ignore them, making them worse than useless. In all cases, the user's work is sacrosanct. Nothing your application does should lose or destroy user's work without explicit user action. Among other techniques, this can be achieved by autosaving, backups of documents, and allowing multiple levels of undo.

9. PROVIDE DIRECT MANIPULATION

Wherever possible, allow users to act on objects and data directly, rather than through dialogs or explicit commands. For example, it is more intuitive to drag a circle object around in a diagram rather than selecting a "Move" command from a menu while the circle is selected. Similarly, in an email application, allow the user to attach files by dragging them from the file manager and dropping them onto the message composition window if they wish.

4. Information Services and Technology (IS&T) – MIT Usability Group: Usability Guidelines

Kilde: <http://web.mit.edu/ist/usability/usability-guidelines.html#nav>

Navigation

1. Current location within the site is shown clearly.
2. Link to the site's main page is clearly identified.
3. Major/important parts of the site are directly accessible from the main page.
4. Site map is provided for a large, complex site.
5. Easy to use Search function is provided, as needed.

Functionality

6. Site accommodates novice to expert users.
7. Functions are clearly labelled.
8. Essential functions are available without leaving the site.
9. Plug-ins are used only if they add value.

User Control

10. Site reflects user's workflow.
11. User can cancel any operation.
12. Clear exit point is provided on every page.

13. Per-page size is less than 50K, to accommodate slow connections.
14. All appropriate browsers are supported.

Language and Content

15. Important information and tasks are given prominence.
16. Information of low relevance or rarely used information is not included.
17. Related information or tasks are grouped:
 - on the same page or menu
 - in the same area within a page
18. Language is simple, without jargon.
19. Paragraphs are brief.
20. Links are concise, expressive, and visible – not buried in text.
21. Terms are defined.

Online Help and User Guides

22. Site is designed to require minimal help and instructions.
23. Help and instructions, if needed, are easily accessible.

System and User Feedback

24. It is always clear what is happening on the site – visual hints, etc.
25. Users can receive email feedback if necessary.
26. Users can give feedback via email or a feedback form.
27. Confirmation screen is provided for form submittal.
28. All system feedback is timely.
29. Users are informed if a plug-in or browser version is required.
30. Each page includes a "last updated" date.

Web Accessibility

W3C's Tools, Checklist, and Guidelines.

Consistency

31. The same word or phrase is used consistently to describe an item.
32. Link reflects the title of the page to which it refers.
33. Browser page title is meaningful and reflects main page heading.

Error Prevention and Correction

34. Users can rely on recognition, not memory, for successful use of the site.
35. Site tolerates a reasonable variety of user actions.
36. Site provides concise instructions for user actions, including entry format.
37. Error messages are visible, not hidden.
38. Error messages are in plain language.

39. Error messages describe actions to remedy a problem.
40. Error messages provide a clear exit point.
41. Error messages provide contact details for assistance.

Architectural and Visual Clarity

42. Site is organized from the user's perspective.
43. Site is easily scannable for organization and meaning.
44. Site design and layout is straightforward and concise.
45. Site design and layout are redundant only when required for user productivity.
46. White space is sufficient; pages are not too dense.
47. Unnecessary animation is avoided.
48. Colours used for visited and unvisited links are easily seen and understood.
49. Bold and italic text is used sparingly.

5. W3C: Web Content Accessibility Guidelines

Kilde: <http://www.w3.org/TR/WAI-WEBCONTENT/>

1. PROVIDE EQUIVALENT ALTERNATIVES TO AUDITORY AND VISUAL CONTENT.

Provide content that, when presented to the user, conveys essentially the same function or purpose as auditory or visual content.

2. DON'T RELY ON COLOUR ALONE.

Ensure that text and graphics are understandable when viewed without color.

3. USE MARKUP AND STYLE SHEETS AND DO SO PROPERLY.

Mark up documents with the proper structural elements. Control presentation with style sheets rather than with presentation elements and attributes.

4. CLARIFY NATURAL LANGUAGE USAGE

Use markup that facilitates pronunciation or interpretation of abbreviated or foreign text.

5. CREATE TABLES THAT TRANSFORM GRACEFULLY.

Ensure that tables have necessary markup to be transformed by accessible browsers and other user agents.

6. ENSURE THAT PAGES FEATURING NEW TECHNOLOGIES TRANSFORM GRACEFULLY.

Ensure that pages are accessible even when newer technologies are not supported or are turned off.

7. ENSURE USER CONTROL OF TIME-SENSITIVE CONTENT CHANGES.

Ensure that moving, blinking, scrolling, or auto-updating objects or pages may be paused or stopped.

8. ENSURE DIRECT ACCESSIBILITY OF EMBEDDED USER INTERFACES.

Ensure that the user interface follows principles of accessible design: device-independent access to functionality, keyboard operability, self-voicing, etc.

9. DESIGN FOR DEVICE-INDEPENDENCE.

Use features that enable activation of page elements via a variety of input devices.

10. USE INTERIM SOLUTIONS.

Use interim accessibility solutions so that assistive technologies and older browsers will operate correctly.

11. USE W3C TECHNOLOGIES AND GUIDELINES.

Use W3C technologies (according to specification) and follow accessibility guidelines. Where it is not possible to use a W3C technology, or doing so results in material that does not transform gracefully, provide an alternative version of the content that is accessible.

12. PROVIDE CONTEXT AND ORIENTATION INFORMATION.

Provide context and orientation information to help users understand complex pages or elements.

13. PROVIDE CLEAR NAVIGATION MECHANISMS.

Provide clear and consistent navigation mechanisms – orientation information, navigation bars, a site map, etc. – to increase the likelihood that a person will find what they are looking for at a site.

14. ENSURE THAT DOCUMENTS ARE CLEAR AND SIMPLE.

Ensure that documents are clear and simple so they may be more easily understood.

6. The Principles of Universal Design

Kilde: http://www.design.ncsu.edu/cud/about_ud/udprinciples.htm

Principle One: Equitable Use

The design is useful and marketable to people with diverse abilities

GUIDELINES

- Provide the same means of use for all users: identical whenever possible; equivalent when not.
- Avoid segregating or stigmatizing any users.
- Provisions for privacy, security, and safety should be equally available to all users.
- Make the design appealing to all users.

Principle Two: Flexibility in Use

The design accommodates a wide range of individual preferences and abilities.

GUIDELINES

- Provide choice in methods of use.

- Accommodate right- or left-handed access and use.
- Facilitate the user's accuracy and precision.
- Provide adaptability to the user's pace.

Principle Three: simple and intuitive

Use of the design is easy to understand, regardless of the user's experience, knowledge, language skills, or current concentration level.

GUIDELINES

- Eliminate unnecessary complexity.
- Be consistent with user expectations and intuition.
- Accommodate a wide range of literacy and language skills.
- Arrange information consistent with its importance.
- Provide effective prompting and feedback during and after task completion.

Principle Four: Perceptible Information

The design communicates necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities.

GUIDELINES

- Use different modes (pictorial, verbal, tactile) for redundant presentation of essential information.
- Provide adequate contrast between essential information and its surroundings.
- Maximize "legibility" of essential information.
- Differentiate elements in ways that can be described (i.e., make it easy to give instructions or directions).
- Provide compatibility with a variety of techniques or devices used by people with sensory limitations.

Principle Five: Tolerance for Error

The design minimizes hazards and the adverse consequences of accidental or unintended actions.

GUIDELINES

- Arrange elements to minimize hazards and errors: most used elements, most accessible; hazardous elements eliminated, isolated, or shielded.
- Provide warnings of hazards and errors.
- Provide fail safe features.
- Discourage unconscious action in tasks that require vigilance.

Principle Six: Low Physical Effort

The design can be used efficiently and comfortably and with a minimum of fatigue.

GUIDELINES

- Allow user to maintain a neutral body position.
- Use reasonable operating forces.
- Minimize repetitive actions.
- Minimize sustained physical effort.

Principle Seven: Size and Space for Approach and Use

Appropriate size and space is provided for approach, reach, manipulation, and use regardless of user's body size, posture, or mobility.

GUIDELINES

- Provide a clear line of sight to important elements for any seated or standing user.
- Make reach to all components comfortable for any seated or standing user.
- Accommodate variations in hand and grip size.
- Provide adequate space for the use of assistive devices or personal assistance.