



OpenUsability.org

where open source meets usability

User Interface Guidelines - Möglichkeiten und Grenzen

Ellen Reitmayr
relevante AG

World Usability Day --- Berlin --- 3. November 2005

Übersicht

- 1 Einführung
 - Wozu User Interface Guidelines?
 - Arten an User Interface Guidelines
 - Beispiele für User Interface Guidelines
 - Grenzen von User Interface Guidelines

- 2 Podiumsdiskussion

1 Einführung

- Wozu User Interface Guidelines?
- Arten an User Interface Guidelines
- Beispiele für User Interface Guidelines
- Grenzen von User Interface Guidelines

2 Podiumsdiskussion

User Interface Guidelines ...

... bilden eine Basis für Design-Entscheidungen.

Sie fördern die Konsistenz bezüglich

- Look and Feel und
- grundlegender Interaktions-Abläufe,

um dem Nutzer die Orientierung innerhalb der Anwendung und des gesamten Arbeitskontextes zu erleichtern.

Zielgruppen

User Interface Guidelines sind vielfältig nutzbar.

Entwickler

Anweisungen, wie das Interface gestaltet werden soll.

- Reduktion des kognitiven Aufwands beim Entwerfen von Oberflächen,
- konsistenter Einsatz von Begriffen und UI Elementen,
- Minimierung der “Meinungs-basierten” Entwicklung.

Zielgruppen

User Interface Guidelines sind vielfältig nutzbar.

Designer

Designvorlagen, Definition von Farbpaletten, etc.

- Konsistenter Einsatz von Begriffen und UI Elementen,
- Minimierung der “Meinungs-basierten” Entwicklung.

Usability-Experten

Heuristiken, die in einem bestimmten Kontext verbindlich sind.

- Evaluationsgrundlage bei der Evaluation von Anwendungen,
- Ableiten von Checklisten.

Zielgruppen

User Interface Guidelines sind vielfältig nutzbar.

Team Leads und Manager

Überblick über Usability und "Usability-Standard".

- Einplanen von Usability in den Entwicklungsprozess,
- Priorisierung von Aufgaben,
- Reduzierte Entwicklungskosten durch bessere Nutzung der Ressourcen.

1 Einführung

- Wozu User Interface Guidelines?
- Arten an User Interface Guidelines
- Beispiele für User Interface Guidelines
- Grenzen von User Interface Guidelines

2 Podiumsdiskussion

Anwendungsbereiche

UI Guidelines können für vielerlei Anwendungsbereiche entwickelt werden:

- Webseiten
- PC-Anwendungen
- Webanwendungen
- Anwendungen für Mobile Devices
- Formatvorlagen
- Notifikationen

aber auch:

- Hardware (Computer, Küchengeräte, ...)

Gültigkeit

Je nach Kontext haben Guidelines eine unterschiedliche Gültigkeit:

Verbindlich für alle (nach geltendem Recht)

- DIN EN ISO 9241-10 bis -17
- Accessibility Guidelines

Für eine Desktop Environment

- Windows
- Apple
- Gnome
- KDE

Gültigkeit

Je nach Kontext haben Guidelines eine unterschiedliche Gültigkeit:

Für einen Entwicklungsbereich, eine Firma o.ä.

- Guidelines für Webanwendungen
- Guidelines für WAP-Seiten
- Guidelines für alle Adobe Produkte

Für eine einzige Webseite, Software o.ä.

- Styleguides für spezifische Webseiten
- Styleguide für YAST

Inhaltsbereiche

UI Guidelines für Software können die folgenden Bereiche umfassen:

- Allgemeine Usability-Prinzipien
- Eingabemöglichkeiten
- Wahl von UI Elementen
- Layout und Design
- Menügestaltung
- Feedback
- Sprache
- Accessibility
- ...

Praktische Anwendbarkeit

Die meisten Application Guidelines bieten:

- Veranschaulichen der Guidelines anhand von Screenshots
- Gründe für Erklärungen (Rationale)
- Positiv- und Negativ-Beispiele, oft mit Screenshots
- Implementierungs-Hinweise

1 Einführung

- Wozu User Interface Guidelines?
- Arten an User Interface Guidelines

- Beispiele für User Interface Guidelines

- Grenzen von User Interface Guidelines

2 Podiumsdiskussion

Bekannte UI Guidelines

Bekannte freie Guidelines im Web- und Software-Bereich:

Web Design

- Jakob Nielsen's Ten Usability Heuristics

Application Design

- DIN EN ISO 9241-10 bis -17
- Apple Human Interface Guidelines
- Windows User Experience Guidelines
- Gnome Human Interface Guidelines
- Java Look and Feel Design Guidelines

Ten Usability Heuristics

Quelle:
www.useit.com
Nov. 2005

by *Jakob Nielsen*

These are ten general principles for user interface design. They are called "heuristics" because they are more in the nature of rules of thumb than specific usability guidelines.

Visibility of system status

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

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.

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.

Consistency and standards

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

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.

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

3.6 Fehlertoleranz

Ein Dialog ist fehlertolerant, wenn das beabsichtigte Arbeitsergebnis trotz erkennbar fehlerhafter Eingaben entweder mit keinem oder mit minimalem Korrekturaufwand seitens des Benutzers erreicht werden kann.

Empfehlungen:

Das Dialogsystem sollte den Entwickler dabei unterstützen, Eingabefehler zu entdecken und zu vermeiden.

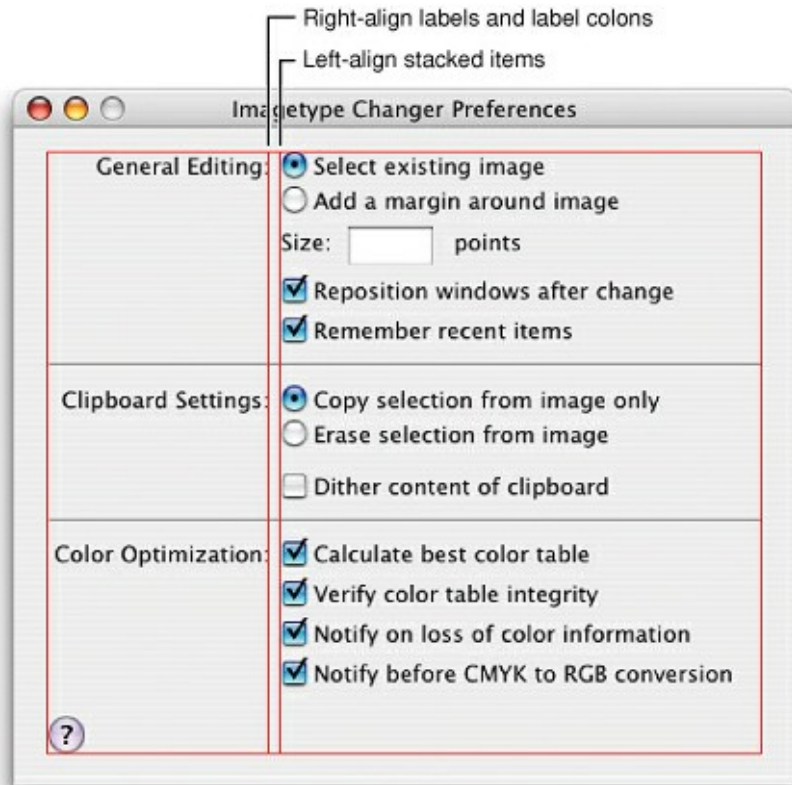
Das Dialogsystem sollte verhindern, daß irgendeine Benutzereingabe zu undefinierten Systemzuständen oder zu Systemabbrüchen führen kann.

Mögliche Beispiele

Wenn eine Folge von Handlungen erforderlich ist, ist das Dialogsystem so gestaltet, daß der jeweils nächste Schritt in einem Ablauf aus den angezeigten Informationen erkannt werden kann. Beispiel: In einem Dialog mit Bildschirmformularen ist die Feldbezeichnung im nächsten auszufüllenden Feld eindeutig lesbar.

Das Dialogsystem prüft die Eingaben auf Ziffern, falls nur Zahlen erlaubt sind.

Figure 15-3 Alignment of labels and controls in a preferences dialog



Quelle:
Apple OSX Human Interface Guidelines
Chapter 19
Nov 2005

There are specific spacing guidelines to keep in mind when designing dialogs. Although "Controls" gives some guidelines for how to arrange controls in relation to each other, in this chapter you can see how these controls should be arranged in relation to the window that contains them. Following are some guidelines that are easy to observe in Figure 15-4:

- Controls not in a group box or a tab view should be 14 pixels from the title bar.
- There should be a 20 pixel margin all the left side, right side, and bottom of a dialog.
- For full-size controls, leave 8 pixels of space between controls.
- Leave at least 12 pixels of space above and below separators.
- Leave at least 16 pixels of space between the bottom group of controls and the buttons (the example in Figure 15-4 has 18 pixels)

1 Einführung

- Wozu User Interface Guidelines?
- Arten an User Interface Guidelines
- Beispiele für User Interface Guidelines

- Grenzen von User Interface Guidelines

2 Podiumsdiskussion

Detailgrad und Anwendbarkeit

Der Detailgrad bestimmt die Anwendbarkeit für Usability-unerfahrene Entwickler.

Probleme:

- Konkrete Richtlinien erzeugen Guidelines mit großem Umfang
→ schwer lesbar!
- Konkrete Richtlinien können nicht alle denkbaren Fälle abdecken.

Plattformabhängigkeit

Guidelines stellen meist nur Empfehlungen für bestimmte Plattformen dar.

Probleme

- Entwicklung von Anwendungen für mehrere Plattformen.

Standard-Lösungen

Guidelines liefern immer nur Standard-Lösungen und können ein maßgeschneidertes Interaktions-Design nicht ersetzen.

Probleme

- Konkrete Anwendungsfälle werden nicht abgebildet.
- Guidelines sind aus Entwicklersicht nicht Aufgaben-orientiert.
- Entscheidungshilfe nur für “kleinere” Designentscheidungen.

2 Podiumsdiskussion

Prof. Dr. Jochen Prümper
FHTW

Christian Theune
Plone, gocept

Sven Neumann
GIMP, jpk

Dr. Siegfried Olschner
SUSE Linux

Ellen Reitmayr
KDE, relevante