User Interfaces

Schnittstelle zwischen Mensch und Maschine:

* Gefährliche Operationen wie löschen oder zurücksetzen
* Befehle sollten eindeutig hervorgehen
* Nicht komplex zu bedienen

Semantisches Netz:

* Anhand von Bildern(eindeutige Aktion) die Bedeutung darstellen

Auf Erfahrung basierendes Netz:

* Bsp.: Gabel zum Essen in der Hand

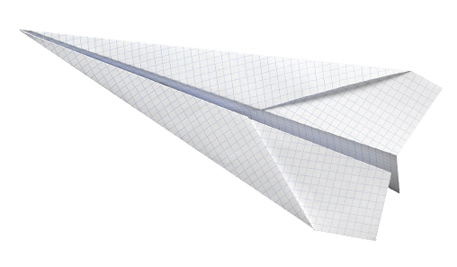
Metaphern:

* Ein Beispiel aus dem realen Leben verwenden und ihr eine neue Bedeutung geben, z.B.: offene Tür zum beenden des Programms.
* Menü (im gastronomischen Sinn):
  + Auswahl von Speisen
* Menü (im technischen Sinn):
  + Auswahl von Operationen

Übungen:







Schaltfläche:

In größeren Kontrollräumen hat man auch viele Knöpfe bzw. ein Kontrollzentrum sog. Schaltflächen.

Interagieren mit User Interfaces

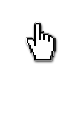
(Systemorientierte) Aktion-Objekt-Interaktion:

* Aktion und Objekt (Bsp.: Mit Handy schreiben und auf SENDE drücken um zu senden, man macht etwas um etwas zu erlangen)

„Objekt-Aktion-Interaktion“:

* Auf ein Bestehendes Objekt können Aktionen ausgeführt werden.

Interaktionsmöglichkeiten:

* Cursor (mit Mouse bewegen)
* Tastatur
* Kommandozeile (CLI)
* Formulare (GUI)
* Drag & Drop (= Mouseinteraktion)
* Stimme
* Buttons, Check buttons, Radio buttons, usw usf.
* Menü mit Befehlen

Beispiele für Aktion-Objekt-Interface?

* Passworteingabe
* Menüleiste
* Wenn man kein definiertes Objekt hat

Shortcut?

* Tastenkombination um eine bestimmte Eingabe zu machen(STRG+C für copy usw.)

Was heist CLI?

* Command Line Interface ist eine Schnittstelle, die Befehle mittels Kommandozeile entgegen nimmt. = Aktion-Objekt

Wie sind GUIs? (A-O oder O-A)?

* Objekt Aktion

Formulare

Was muss ein Formular können:

* Wissen was man eingeben muss/kann
* Konvention wie das auszufüllen ist muss bekannt sein (alles ausfüllen usw.)
* Den User helfen mit Eingaben
  + Z.B.: Datum mittels Kalender auswählen usw.
  + Kurze Klärung der Felder
  + Die Grenzen der Felder (positive Zahlen oder nicht größer als 50…)
  + Warnungen bei falscher Ausfüllung (direkt bei Eingabe überprüfen, nicht am Ende)
  + Pflichtfeld ?
  + Links oder rechts justieren? Zahlen rechts und Schriften links
  + Löschen/überschreiben/usw. soll gegeben sein
  + Bei Ende des Formulars soll der Benutzer wissen wofür man es braucht.
  + Größere Formulare(länger wie eine Seite) sollte man nicht überschreiten
  + Sollte das Formular doch zu groß sein, sollte man es aufteilen

Menü

Textuelles MENU (CLI wie bei DVD-Verwaltung)

Graphisches MENU (Wie bei Websiten):

* Einheitsmenü (nur eine Menu ohne Verzweigungen)
* Baumstruktur (Verzweigungen unter den einzelnen Menüpunkten)
* Netzstruktur (man gelangt durch Überspringen eines Untermenüs in ein gewünschtes Verzeichnis)
* Farbliche Unterscheidung zwischen Punkten.
* POP-UP-Menü = Objekt – Aktion
* Menübalken
  + Balken mit Menübalken 🡪 hat Untermenüs in denen Aktionen enthalten sind
* Implizites Menü:
  + Objekt 🡪 Aktion (pop-up-menü)
* Explizites Menü:
  + Aktion 🡪 Objekt
* Untermenü
  + Pfeil zur Erkennung
  + Grau oder Schwarz(kann man auswählen)

Dialoge/ Interaktionsobjekte

Definition: Objekte die im Interface vorkommen, zum Interagieren mit dem System

Bsp.: Checkbox(hängen nicht zusammen), Radiobutton(man kann nur eines auswählen),…

* Auswahlfenster:
  + Fenster wo Objekte enthalten sind (z.B.: wo man font-fam auswählt)
* Auswahlleiste:
  + Leiste mit Bildern(meistens) wo man Objekte auswählen kann(z.B.: Überschriftenauswahl)
* Bei hover sollte überall stehen was der Befehl macht, die leiste/Fenster sollte verschiebbar sein
* Dialog
  + Modale Dialogfenster:
    - Z.B.: Login, hat nichts mit der Fkt. des Prg. zu tun, man braucht das nur um das Prg. zu nutzen
  + Nicht modale Dialogfenster:
    - Hat etwas mit dem Programm zu tun, z.B.: Datei Öffnen um ins Prg zu laden
* Informative Dialogfenster:
  + Man muss schauen ob man sie modal oder nichtmodal verwendet, hello world🡪nicht modal
* Interaktive Dialogfenster:
  + Benutzer muss mit System aktiv interagieren z.B.:
* Warnungen:
  + Das System läuft weiter, es ist ein Hinweis, dass man das Problem lösen sollte, z.B.: zu wenig Speicher
* Fehler:
  + Da ist es schon zu spät, das Programm kann nicht mehr weiterlaufen
    - Es ist wichtig dass bei Fehlermeldungen dabei steht, was man machen soll
* Was ist ein Radiobutton:
  + Mehrere Auswahlmöglichkeiten, mit Radiobutton kann man nur eine auswählen

Begriffe

Usability: Die Gebrauchstauglichkeit ist die Eignung eines Gutes im Hinblick auf die Verwendung.

Die Aufmerksamkeit Konzentriert sich nur auf ein Objekt. Objekte werden gleichzeitig wahrgenommen, aber nicht registriert.

Gruppenbildung von Sinneinheiten (z.B.: Punkt bei Tausenderstelle)

Nur 7 unabhängige Informationseinheiten können verarbeitete werden(nicht mehr als 7 Auswahlmöglichkeiten)

Elemente fallen auf, wenn sie eine Serie unterbrechen

Wahrnehmung

Genuines Wissen:

* Angeborenes Wissen hilft den Menschen sich zu orientieren
* Ermöglicht Wahrnehmungvo Strukturen

Prägnanz:

* Ein grob wahrgenommenes Bild wird so einfach wie möglich interpretiert/ verfollständigt.

Zahlen die wichtig sind sollen hervorgehoben werden

Bei formen erregt Aufmerksamkeit:

* Farbänderung
* Position
* Helligkeit
* Größe
* Drehung…

Lesbarkeit

Typografie:

Darstellung von Texten in digitalen Medien bedarf einer gerechten Anwendung

* Auflösung eines Displays und die damit verbundenen Einschränkungen

Man sollte Standardschriftarten verwenden um die Lesbarkeit zu steigern

Serifenlose Schrift

Serienschrift

Maximalkontrast eher abschwächen

Geglättet: Hintergrund und Schriftart wird angepasst, damit bei Schriften die Pixel nicht extrem erkennbar sind

Ungeglättet: Man erkennt einzelne Pixel bei Schriften usw.

Glätten um Lesbarkeit zu Steigern

Textausrichtung:

* Normaler Text sollte linksbündig sein.
* Sonst nach Land/Daten/Gewohnheit ausrichten

Groß- und Kleinschreibung:

* Es ist leichter lesbar Klein und Groß zu schreiben
* Nicht alles groß schreiben

Viel Text in logische Absätze aufteilen

Piktogramme

Durch Bilder/icons/symbole soll man Metaphern einsetzen

Farben

Sinnvolle Verwendung

Nie mehr wie 5 Farben gleichzeitig

Viele Menschen sind Farbenblind deshalb nicht nur farbliche Unterscheidung, auch durch unterstreichen oder fett drucken

Raster

Konventionen: Navigation links, Titel oben,…

Konsistenz im Interface ermöglicht einfache Identifizierung

Usability

Gebrauchstauglichkeit: Setzt sich aus Effektivität, Effizienz und Zufriedenheit zusammen.

Was ist eigentlich schön

Der goldene Schnitt

Verhältnis 1,6 und 1

Charakteristiken einer guten Software

Hohe Performance:

User Interface muss immer flüssig wirken, egal wie aufwändig die Hintergrundprozesse sind, das Interface darf nie ins Stocken geraten.

In verschiedenen Ländern sind verschiedene Datenübertragungsraten

Bei Google ist der Suchalgorithmus viel schneller als Yahoo

Ease of Use (Einfachheit der Verwendung):

* Einfach zu bedienen und schnell (Apple)

Anpassungsfähigkeit:

* Anpassung Landesabhängig

Attraktives Aussehen

* Ist eines der wichtigsten Kriterien weil viele Kunden nur auf das Aussehen Wert legen

Mobilität:

* Website für Handy und PC

Entwicklung der Seite

Business Objekt: Daten die auf der Datenbank liegen  
Datenintegrität gewährleisten, Datenschutz

Für ein Business Objekt gibt es mehrere Interfaces Objekte.

Zugriffsberechtigung muss geprüft werden.

Man kann mit einem Interface Objekt ein anderes aktivieren usw.

Anhand des use-case-Diagramms, kann man die Website erstellen & die Navigtionsstruktur

Bei Projektstart ist es wichtig das Vorgehensmodell zu wählen und jeden Schritt dokumentieren.

Priorizing design desicions?????