

Grundlagen

Weblayout

- orientiert sich am Printdesign
- Unterschied zum Printdesign: Ausgabeformat (=Bildschirmauflösung) nicht feststehend, daher muss man entweder selbst Standard festlegen (z.B. 1024 x 768 Pixel) oder im Design flexibel bleiben

- Elemente einer Website:
 - Kopfbereich (head) - oberer Bereich für Bannergrafiken und Navigationselemente
 - Navigationselemente - gruppiert (z.B. Serviceangebote, Index, usw.)
 - Inhalte (content): Texte, Bildergalerien, usw.
 - Fußzeile (footer) - unterer Bereich für Navigationselementen und Copyright/Disclaimer

- Allgemeine Regeln:
 - im Layout (Farben, Schriften, Gestaltungselemente) konsequent bleiben
 - jede Information mit 3 Klicks erreichbar
 - maximal 7 Navigationselemente je Gruppe
 - Farben: "Weniger ist mehr" (2-3 Grundfarben + hell/dunkel Abstufungen)
 - Wirkung und Bedeutung von Farben beachten

- Wahrnehmung und Gestaltpsychologie:
 - Prägnanz ("Gute Gestalt"): zuerst werden einfache geometrische Formen wahrgenommen, dann die Details
 - Erfahrung: Wiedererkennen vertrauter Formen
 - Negativraum ("Figur und Grund"): Kontext bestimmt Wahrnehmung (Kippbilder)
 - Kontrast: Auffälliges springt ins Auge
 - Nähe: Zusammenstehendes gehört zusammen (=Gegensätze klar abgrenzen)
 - Gleichheit: Ähnliches erscheint als Gruppe (Raum-Zeit-Kontinuität)
 - Einheit und Harmonie: Eindruck visueller Verbindung schafft Gruppeneindruck
 - Geschlossenheit: Auge ergänzt Fehlendes bei angedeuteten Formen

- Regeln für Formataufteilung:
 - Goldener Schnitt (Φ)
 - arithmetische Folge (konstantes Intervall)
 - geometrische Folge (Potenzierung)

- relative Anordnung von Gestaltungselementen:
 - Position Elemente-Format: zentriert vs. optische Mitte, erhaben vs. bodenständig, ausgewogen vs. unausgewogen, symmetrisch (achssymmetrisch, punktsymmetrisch) vs. asymmetrisch
 - Position Elemente-Elemente: einfach gereiht, rhythmisch, gerastert, geometrisch, gestreut, verdichtet, gruppiert, gestaffelt

- Typografie und Webseiten:
 - Nicht mehr als 2 Schriftarten (Vorsicht vor exotischen Schriftarten!)
 - Fließtextgröße (Mengentext): 9-12 pt, Schaugrößen (Überschrift): ab 14pt; Mengentext von Überschrift deutlich unterscheidbar
 - Zeilenvorschub mind. 1,2faches der Schriftgröße
 - hoher Kontrast zwischen Schriftfarbe und Hintergrund
 - linksbündiger Flattersatz

Modul	→ Gestaltung und Betreuung dynamischer Webseiten mit HTML (inkl. CSS) & PHP (inkl. MySQL)
Termin	→ 27.10. bis 21.11.2008.
Dozent	→ Uwe Hartwig, Halle (Saale)

- Typografische Gestaltungselemente

- Absätze vs. Leerräume
- Linien vs. Flächen
- Icons vs. Hintergrundgrafiken

HTML

- legt Struktur eines HTML-Dokumentes fest
- umfasst aktuell >70 Elemente ("Tags") plus obligatorische und fakultative Attribute-Wert-Paare (abhängig von HTML-Version)
- erfordert verschachtelte Strukturen, die festlegen, welche child-Elemente in einem parent-Element enthalten sein dürfen oder müssen (Document Tree)
- leere Elemente umschließen keine Textdaten und können nicht verschachtelt werden (laden z.B. Bilder oder Skripte)
- Aufbau:
 - `<elementname attribut_1="wert_1" ... attribut_n="wert_n">... </elementname>`
 - Aufbau leeres Element: `<elementname />`

CSS

- Regelanweisungen für Bildschirmausgabe von HTML-Elementen
- aktueller Standard umfasst >100 Eigenschaften plus Wertangaben, jedoch sind Eigenschaften und Werte nicht beliebig mit HTML-Elementen kombinierbar
- grundlegende Eigenschaft: Anzeige/display (z.B. Fließtext, Blockelement)
- "kaskadieren": CSS-Regeln werden je nach Gewicht auf HTML-Element angewendet und vererbt (an childs/descendants im Document Tree)
- Eigenschaftsklassen:
 - Box-Modell (z.B. Ränder und Abstände)
 - Farben und Hintergründe (z.B. Hintergrundbilder)
 - Schrift (z.B. Schriftgröße)
 - Textformatierung (z.B. Blocksatz)
- CSS-Regeln stehen:
 - direkt bei HTML-Elementen
 - im head-Bereich eines HTML-Dokuments
 - in separaten .css-Dateien
- Regel-Aufbau: `Selektortyp { eigenschaft_1:wert_1;...;eigenschaft_n:wert_n}`
- Selektor: verbindet Eigenschaft-Wert-Paare mit HTML-Elementen
- Selektortypen:
 - Tag-Selektor
 - Klassen-Selektor
 - ID-Selektor
 - Kontext-Selektor
- Implementation von CSS-Standards variiert in der Realität je nach Browsertyp- und Version

Javascript

- auf Client(=Webbrowser) zugeschnittene Programmiersprache
- läuft clientseitig im Sicherheitskäfig ("Sandbox"-Verfahren)

Modul	→ Gestaltung und Betreuung dynamischer Webseiten mit HTML (inkl. CSS) & PHP (inkl. MySQL)
Termin	→ 27.10. bis 21.11.2008.
Dozent	→ Uwe Hartwig, Halle (Saale)

- Objekte, Unterobjekte und deren Methoden werden mit Punktoperator verbunden: `objekt.unterObjekt.methode(parameter_1,...,parameter_n) = "zugewieseneWerte"`
- setzt Objekte als Prototypen (Standardobjekte) ein
- Hierarchie der Standardobjekte:
 - oberstes Objekt "window":
stellt Methoden bereit, um Browserfenster anzusprechen (z.B. Bildschirmbreite) und alles, was darin enthalten ist (also auch HTML-Dokumente)
 - nächstes Objekt "document" ("window.document"):
stellt Methoden bereit, um Elemente des HTML-Dokuments anzusprechen und beinhaltet weitere Objekte für Klassen von HTML-Elementen (z.B. Bilder, Formulare);
 - in der Praxis werden hauptsächlich Standard-Objekt und Unter-Objekte verwendet
- Skripte werden über Eventhandler angesteuert
- Eventhandler stehen bei HTML-Elementen:
`<html-element event="javascriptfunktion(parameter_1,...,parameter_n)">`
- wichtige Eventhandler:
 - onmouseover (Maus steht über HTML-Element)
 - onmouseout (Maus verlässt HTML-Element)
 - onmousemove (Maus bewegt sich im Browserfenster)
 - onclick (Maus klickt auf HTML-Element)
 - onfocus (HTML-Element angeklickt oder mit Tab aktiviert)
 - onblur (HTML-Element nicht mehr angeklickt / aktiviert)
 - onsubmit (Formulardaten sollen abgeschickt werden)
 - onload/onunload (Dokument wird geladen/verlassen)
- Javascripte stehen:
 - bei Eventhandlern
 - innerhalb des script-Tags (`<script>...</script>`) in HTML-Dateien
 - in separaten .js-Dateien
- Erstellen eigener Klassen:
 - Definition - `function meineKlasse () {...}`
 - Instanz - `ersteInstanz = new meineKlasse()`
- Implementation von Javascript und DOM variiert je nach Browsertyp- und Version

DHTML

- verleiht statischen HTML-Dokumenten nachträglich Interaktivität
- verändert HTML-Elemente/CSS-Regeln je nach Anwenderereignis (event)
- Notation für Zugriff auf HTML-Elemente mit Element-ID:
 - HTML: `<elementname id="bezeichner" attribut="wert">`
 - JS: `document.getElementById("bezeichner").attribut = "wert"`
- Notation für Zugriffe auf CSS-Eigenschaft eines HTML-Elements mit Element-ID:
 - CSS: `element#bezeichner { css-eigenschaft:wert }`
 - JS: `document.getElementById("bezeichner").style.cssEigenschaft = "wert"`

PHP

- serverseitige Programmiersprache = unabhängig von Browsern
- produziert HTML-Code nach vorgegebenen Parametern aus PHP-Anweisungen
- stellt reichhaltige Bibliotheken zur Verarbeitung von HTML-Code, Nutzereingaben (regular expressions) und MySQL-Anfragen bereit
- PHP-Anweisungen stehen in php-Dokumenten innerhalb von `<?php ?>`
- Standard-Dateiendung: `.php`, `.tpl`
- arbeitet meist mit HTML-Vorlagen, die Platzhalter für Resultate von PHP-Anweisungen enthalten (Template)

- Ausgabe von HTML über PHP:
 - unmittelbar mit `echo/print` 'HTML-Code'
 - Variable zuweisen (`$var = 'HTML-Code'`), deren Inhalt bei Bedarf ausgegeben wird (`echo $variable`)
 - Unterbrechen eines Blocks von PHP-Anweisungen (`?> HTML-Code <?php`)

- Erscheinungsweise von PHP-Code:
 - ad-hoc Eingabe von Standardfunktionen
 - prozedural in eigenen Funktionen
 - objektorientiert mit Klassen, Klasseneigenschaften und Methoden

- Implementation von OOP variiert je nach Version:

	PHP4	PHP5
Deklaration von Objekteigenschaften	<code>var \$eigenschaft;</code>	<code>public \$eigenschaft; protected \$eigenschaft; private \$eigenschaft;</code>
Konstruktor	<code>class objekt { function objekt() { } }</code>	<code>class objekt { function __construct() { } }</code>
Objektübergabe	als Wert	als Referenz

MySQL

- organisiert Daten atomar, redundanzfrei und konsistent in Tabellen (relationale Datenbank, Normalformen)
- eigenständige, für Webanwendungen angepasste Variante von SQL

- Hauptoperationen über PHP:
 - Datensätze bearbeiten (abfragen/select, hinzufügen/insert, verändern/update, löschen/delete)
 - Informationen abfragen (über Datenbank, Tabelle, Feld, von Abfragen betroffene Datensätze, usw.)

- Beispiel-Syntax für Abfragen:

```
SELECT tabellenfeld_1 AS ausgabefeld_1,...,tabellenfeld_n AS ausgabefeld_n
FROM tabelle
WHERE bedingung1,...bedingungx
ORDER BY tabellenfeld1
LIMIT startsatz,zeilenanzahl
```

Javascript-Frameworks und AJAX

- Javascript-Frameworks: Javascript-Bibliotheken, die DHTML- und AJAX-Funktionalität bereitstellen
- Schritt zu interaktiven, desktopähnlichen Webanwendungen; Grenzen zwischen Online- und Offlineanwendungen verschwinden
- Frameworks bauen teilweise aufeinander auf (z.B. beinhaltet script.aculo.us Prototype-Bibliothek)
- Bibliotheken wie script.aculo.us und Prototype sind frei verfügbar, einige große IT-Hersteller stellen jedoch eigene Frameworks zur Verfügung (z.B. Adobe Spry, Microsoft Ajax, Yahoo User Interface)
- AJAX (Asynchronous Javascript and XML): asynchrone Kommunikation zwischen Server und Client, bei der Struktur-Elemente (HTML, XML) oder Nutzerdaten im Hintergrund vom Server nachgeladen werden
- AJAX ruft im Hintergrund Informationen vom Server ab, ohne das ein neues Dokument geladen wird
- AJAX läuft z.B. auf Basis des HTTP-Request-Objekts:
 - (FF) `http_request = new XMLHttpRequest()`
 - (IE) `http_request = new ActiveXObject("Microsoft.XMLHTTP")`
- zentrale HTTP-Request-Methoden:
 - Datenverbindung herstellen (`open`)
 - Zustand der Verbindung abfragen (`readyState`)
 - Datenformat festlegen (`setRequestHeader`)
 - Daten an Serverskripte senden (`send`)
- Implementation von AJAX baut auf Javascript und DOM variiert daher ebenfalls je nach Browsertyp- und Version

Modul	→ Gestaltung und Betreuung dynamischer Webseiten mit HTML (inkl. CSS) & PHP (inkl. MySQL)
Termin	→ 27.10. bis 21.11.2008.
Dozent	→ Uwe Hartwig, Halle (Saale)

Internet Ressourcen

HTML und CSS

W3C-Referenzen für HTML XHTML 1.0 und CSS 2.1

<http://www.edition-w3c.de/TR/xhtml1>
<http://www.edition-w3c.de/TR/1998/REC-CSS2-19980512/>
<http://www.w3.org/TR/2007/CR-CSS21-20070719>

Valides Webdesign

<http://validator.w3.org/>
<http://jigsaw.w3.org/css-validator/>

selfhtml.org

Umfassendes deutschsprachiges Nachschlagewerk zu HTML, CSS und Javascript
<http://de.selfhtml.org/>

Verbreitung moderner Grafikbrowser

<http://www.w3schools.com/browsers/>

Implementierung von CSS in modernen Browsern

<http://css4you.de/browsercomp.html/standardbrowser/>

Einführung in CSS-Hacks

<http://www.sitepoint.com/article/browser-specific-css-hacks>

Webdesignstandards und Barrierefreiheit

Standards im Web und Acid-Tests

<http://www.webstandards.org/>

CSS Zengarden

Demonstrationen von Webdesign mittels CSS
<http://www.csszengarden.com/tr/deutsch/>

Dr.Web-Magazin rund ums Webdesign

<http://www.drweb.de>

Informationen über barrierefreies Webdesign

<http://www.w3.org/TR/AERT>
<http://www.macx.de/projekte/barrierefrei>

BIENE-Award für Barrierefreie Websites

<http://www.biene-award.de/award/preistraeger/>

Javascript

Javascript-Buch

<http://www.galileocomputing.de/openbook/javascript/>

Dynamicdrive: Javascripte zum Herunterladen und Experimentieren

<http://www.dynamicdrive.com>

DHTML und AJAX

DHTML-Beispiele mit Erläuterungen bei selfhtml.org

<http://de.selfhtml.org/dhtml/beispiele/>

Prototype Javascript Framework

<http://www.prototypejs.org>

script.aculo.us Javascript Framework

<http://script.aculo.us/>
<http://scriptaculous.de>
<http://github.com/madrobby/scriptaculous/wikis/>

AJAX Einführung auf mozilla.org

http://developer.mozilla.org/De/AJAX:Getting_Started

AJAX-Tutorial auf admin-wissen.de

<http://www.admin-wissen.de/tutorials/eigene-tutorials/webentwicklung/ajax-tutorial/>

Modul	→ Gestaltung und Betreuung dynamischer Webseiten mit HTML (inkl. CSS) && PHP (inkl. MySQL)
Termin	→ 27.10. bis 21.11.2008.
Dozent	→ Uwe Hartwig, Halle (Saale)

PHP

PHP 5 Referenz

<http://www.php.net/manual/de/>

PHP-Forum deutsch

<http://php.de>

Dynamic Web Pages - Skriptarchiv

<http://www.dynamic-webpages.de>

pearl-Unity - Scriptarchiv

<http://www.perlunity.de/php/scripts/download.shtml>

PHP Fehlermeldungen

<http://www.php-fehlermeldungen.de/>

Smarty - PHP Template Engine

<http://www.smarty.net/>

Mysql

Mysql 5.1 Referenz

<http://dev.mysql.com/doc/refman/5.1/de/index.html>

Mysql Datenbankhandbuch mit vielen Beispielen

<http://www.little-idiot.de/mysql/>

Tools & Miscellaneous

XAMPP- und XAMPPlite - Entwicklungsumgebung

<http://www.apachefriends.org/de/xampp.html>

MoWeS Portable - Entwicklungsumgebung

<http://www.chsoftware.net/de/useware/mowes/mowes.htm>

Notepad++

<http://notepad-plus.sourceforge.net/de/site.htm>

Notepad++ Plugins

http://sourceforge.net/project/showfiles.php?group_id=189927

Phase5

<http://www.phase5.info/>

Phase 5 Erweiterungen und Plugins

<http://www.qhaut.de/forums/index.php?automodule=downloads>

Mysql-Workbench5

<http://dev.mysql.com/downloads/workbench/5.0.html>

Adobe Flash zur Einführung

http://www.flashtuts.de/htdocs/flash-Tutorials-tutorials_uebersicht.html

Domainfinder - Überprüft Verfügbarkeit von .de-Domains

<http://www.domainfinder.de>

Denic Whois - Informationen über registrierte Domains

<http://www.denic.de/de/whois/>

E-Mail-Header lesen und verstehen

<http://www.th-h.de/faq/headerfaq.php>