

# PROJEKTOWANIE STRON WWW

Wykład I

Paweł Woszkowski  
SWSIM, 2010

# PLAN ZAJĘĆ

- technologie,
- proces powstawania strony www,
- podstawowe zasady,
- xhtml, css,
- projektowanie,
- publikacja, hosting, promocja.

# STRONA WWW

Strona internetowa (strona WWW) - dokument utworzony w którejś z odmian języka SGML (HTML, XML), pobierany z dysku lokalnego komputera bądź serwera internetowego i interpretowany po stronie użytkownika za pomocą przeglądarki.

Cechą charakterystyczną stron WWW jest hipertekstowość i często multimedialność. Część stron internetowych może zawierać w sobie elementy prezentacji stworzonych za pomocą dodatkowych technologii (np. Flash), część tworzona jest dynamicznie w momencie wysłania przez przeglądarkę zapytania do serwera.

Zbiór powiązanych tematycznie i umieszczonych na jednym serwerze stron internetowych często nazywa się serwisem internetowym/WWW lub witryną internetową.

*źródło: wikipedia*

# RODZAJE STRON WWW

- portal informacyjny,
- komercyjna,
- społecznościowa,
- eventowa,
- blog.

# PORTAL INFORMACYJNY

- horyzontalny
  - onet.pl
  - yahoo.com
- wertykalny (wortal)
  - wrower.pl
  - e-teatr.pl

# KOMERCYJNA

- firmowa
  - netizens.pl
- korporacyjna
  - pkobp.pl
- produktowa
  - jbl.pl

# SPOŁECZNOŚCIOWA

- [wykop.pl](http://wykop.pl)
- [filckr.com](http://filckr.com)
- [wikipedia](http://wikipedia)

# EVENTOWA

- [opener.pl](http://opener.pl)
- [comety.viva-tv.pl](http://comety.viva-tv.pl)

# BLOG

- [mojageneracja.pl/](http://mojageneracja.pl/) | 980
- [photoblog.pl/](http://photoblog.pl/)
- [twitter](#) / [blip](#) / [facebook](#)

# TECHNOLOGIE

- client side
  - html, css, javascript, flash
- server side
  - php, asp, jsp, sql, apache, iis

# PROCES POWSTAWANIA STRONY WWW

- strategia (koncepcja),
- kreacja (przetarg, key visuelle),
- architektura informacji (makieta, web usability),
- kreacja (layout wszystkich stron),
- programowanie (silnik strony, cms, server-side scripting).

# PROCES POWSTAWANIA STRONY WWW

- cięcie strony (html, xhtml),
- flash,
- testowanie,
- poprawki,
- hosting, pozycjonowanie, promocja.

# TWORZENIE MAKIETY

- podstawowe narzędzie architekta informacji,
- umożliwia obserwację interakcji na stronie,
- pozwala na wykrycie wielu błędów logicznych,
- ułatwia pracę grafikowi,
- ułatwia pracę z klientem.

# NARZĘDZIA DO TWORZENIA MAKIET

- axure,
- iplotz,
- balsamiq mockups,
- flash, powerpoint, photoshop.

# PODSTAWOWE ZASADY PROJEKTOWANIA

- określenie grupy docelowej,
- wybór technologii,
- dobór rozdzielczości ekranu,
- obsługiwane przeglądarki,
- walidacja.

# WEB USABILITY

- ergonomia interaktywnych urządzeń oraz aplikacji,
  - intuicyjna nawigacja,
  - ułatwienia dostępu do poszukiwanej informacji,
  - zapewnienie zrozumiałej dla użytkownika komunikacji.

# USABILITY TIPS & TRICKS

- tytuły pól w formularzach najlepiej umieszczać nad nimi,
- ludzie skupiają uwagę na twarzach,
- większość (77%) użytkowników nie przewija strony,
- idealne pole do wyszukiwania ma 27 znaków,
- 5 testerów wykryje 85% błędów, 15 - 100%,
- nie zauważamy reklam,

# USABILITY TIPS & TRICKS

- logowanie i rejestracja najczęściej umieszczane są w prawym górnym rogu,
- ponad 90% stron ma stałą szerokość i są wyśrodkowane,
- zasada 3 kliknięć,
- zasada 80/20,
- ludzie nie lubią zmian,

# USABILITY TIPS & TRICKS

- ludzie nie lubią podawać swoich danych,
- ludzie nie lubią długich formularzy.

# XHTML

XHTML - *Extensible Hypertext Markup Language*, czyli Rozszerzalny Hipertekstowy Język Oznaczenia. Reformuje on znane zasady języka HTML 4 w taki sposób, aby były zgodne z XML (*Extensible Markup Language* - Rozszerzalny Język Oznaczenia).

# PRZYKŁADOWA STRUKTURA DOKUMENTU XHTML

```
<?xml version="1.0" encoding="ISO-8859-2"?>
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl">
  <head>
    <title>Tytuł dokumentu</title>
  </head>
  <body>
    <p>Przejdź do <a href="http://example.org/">example.org</a>.</p>
  </body>
</html>
```

# HTML VS XHTML

- Znaczniki należy zamykać obowiązkowo w kolejności odwrotnej do ich otwierania
- Nazwy znaczników i atrybutów obowiązkowo muszą być pisane małymi literami
- Dla elementów niepustych znaczniki zamykające są obowiązkowe
- Wartości atrybutów muszą być zawsze ujęte w cudzysłowy
- Puste elementy muszą mieć znacznik zamykający albo ich znacznik otwierający musi się kończyć na `/>`
- Nie wolno zagnieżdżać (umieszczać jeden w drugim) następujących znaczników: `a`, `pre`, `button`, `label`, `form`
- Atrybut `name` dla znaczników `a`, `applet`, `form`, `frame`, `iframe`, `img`, `map` jest zdeprecjonowany. Należy stosować zamiast niego atrybut `id`

# DOKUMENT

- XHTML - struktura dokumentu.
- CSS - prezentacja,
  - język CSS to reguły opisujące wygląd dokumentu opisanego za pomocą jakiegoś języka znaczników, np. XHTML,
  - reguły CSS gromadzi się w plikach nazywanych arkuszami stylów,
  - reguły mogą się nakładać, czyli być kaskadowe.

# ELEMENTY XHTML

- blokowe

Bloki otaczające tekst, najczęściej są same w linii, mogą zawierać w sobie elementy liniowe i w niektórych przypadkach elementy blokowe.

- liniowe

Elementy wplecione w tekst, dzielone i przenoszone do nowej linii, nie mogą zawierać elementów blokowych.

# ELEMENTY BLOKOWE

- nagłówki `<h1>`, `<h2>`, ..., `<h6>`
- akapity `<p>` i łamanie linii `<br />`
- listy `<ul>`, `<ol>`, `<dl>`
- cytowany fragment `<blockquote>`
- oddzielenie elementów `<hr />`
- tekst preformatowany `<pre>`
- dane kontaktowe `<address>`
- sekcja `<div>`

# ELEMENTY LINIOWE

- cytat `<q>`
- emfaza `<em>` i silna emfaza `<strong>`
- kod `<code>` (często wewnątrz `pre`)
- indeks dolny `<sub>` i górny `<sup>`
- drobny tekst `<small>`
- dowolny zakres `<span>`

# HIPERTEKST

- odnośniki `<a>`  
`<a href="http://www.onet.pl">Wiadomości</a>`
- odnośnik do e-mail  
`<a href="mailto:pawel@woszkowski.com">Wyślij<a>`

# OBRAZKI

```

```

- src - ścieżka
- alt - tekst alternatywny
- title - dymek
- width, height

# CSS

- łączenie xhtml i css

```
<link rel="stylesheet" href="arkusz.css" type="text/css" />
```

- składnia CSS

selektor {właściwość: wartość; inna-właściwość: jakieś wartości;}

kolejny selektor, jeszcze jeden selektor {więcej-właściwości: wartości;}

# SELEKTORY

- elementu / typu  
`li {font-weight: bold;}`
- identyfikatora  
`li#menu {margin: 0;}`      `<li id="menu">lorem</li>`
- klasy  
`div.abc {color: blue;}`      `<div class="abc">lorem</div>`
- pseudoklasy / pseudoelementu  
`a:hover {color: blue;}`  
pseudoklasy: link, visited, hover, focus, active, first-child, lang, not

# SELEKTORY

- ogólny  
\* {color:red;}
- kombinatory
  - potomka div li {color:red;}
  - dziecka div > li {color:red;}

# SELEKTORY

- łączenie i grupowanie
  - h1, h2, h3 {font-weight: bold;}
  - div.abc#dom:hover:first-letter

# JEDNOSTKI

- piksele - px
- procenty - %
- wysokość tekstu - em
- wysokość litery x - ex
- m, mm, in, pt (1/72 cala), pc (12 pt)

# KOLORY

- #0f0
- #00FF00
- `rgb(0, 255, 0)`, `rgb(0, 100%, 0)`, `rgba(0, 100%, 0, 1)`
- `transparent`, `rgb(0, 0, 0, 0)`

# URL

- `url(obrazek.png)`
- `url("http://adres.com/obrazek.jpg")`
- `url('../arkusz.css')`

# TEKST

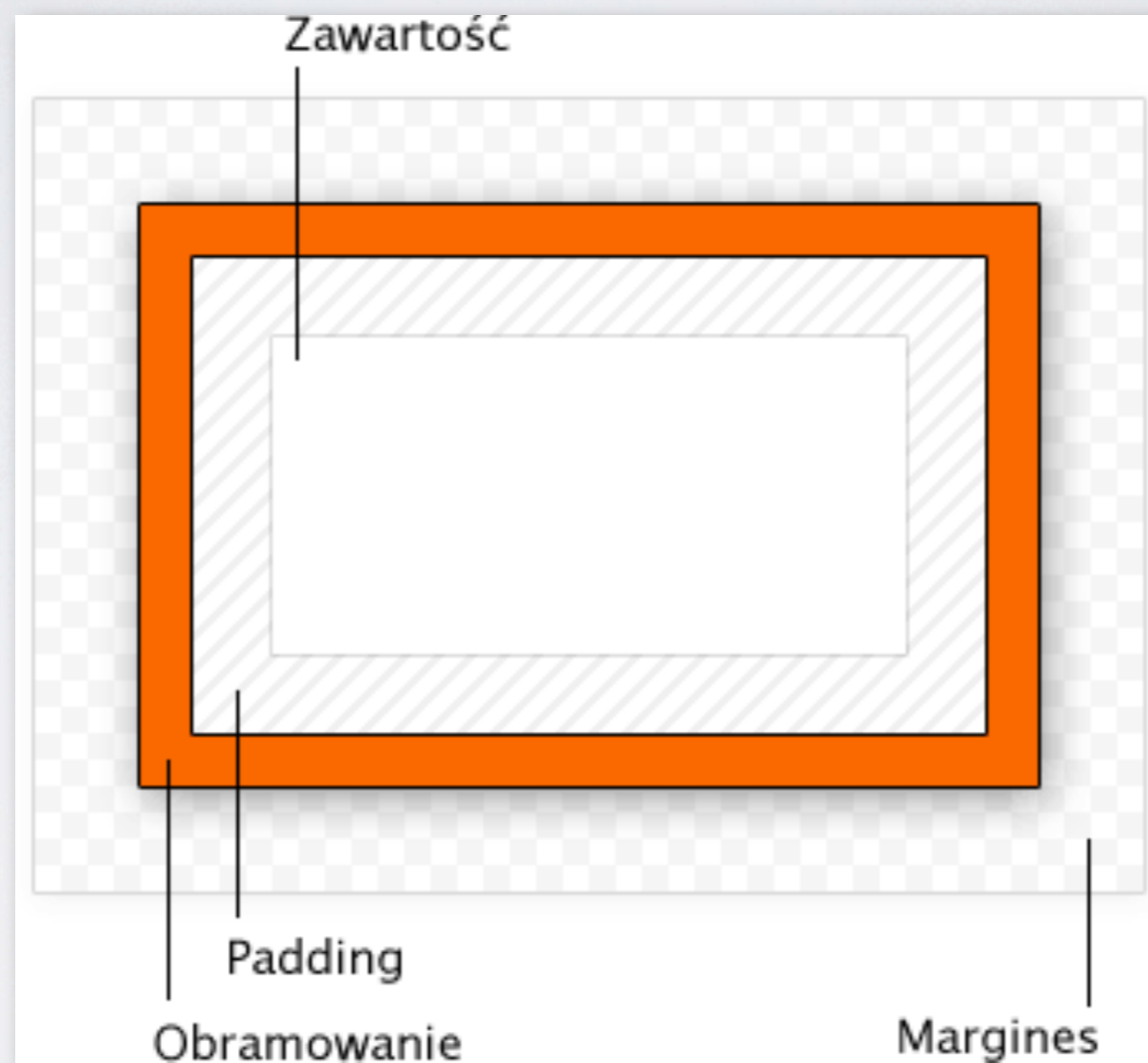
- color
- font-family     font-family: "Avant Garde Gothic Medium", Verdana, Helvetica, sans-serif;
- font-size
- font-weight
- font-style
- letter-spacing, word-spacing, line-height

font: bold 80% sans-serif

# TEKST

- text-decoration (m.in. underline)
- text-align, vertical-align
- text-indent
- text-transform (np. uppercase)
- text-shadow

# BOX MODEL



# WYMIARY

- width, height
- min-width, max-width, min-height, max-height
- overflow: visible, hidden, scroll, auto

overflow:visible

overflow:hid

overflow:au



# KRAWĘDZIE

- top right bottom left
- padding
- margin
- margin:auto - centrowanie
- border

# TLO

- background-color
- background-image
- background-repeat (repeat, repeat-x, repeat-y, no-repeat)
- background-position
- background-attachment (fixed, scroll)

```
background: red url(foo.png) 5px 100% no-repeat;
```

# LISTY

`<ul> <ol> <li>`

- `list-style-type` (none, disc, circle, square)
- `list-style-image`
- `list-style-position` (outside, inside)

`list-style`