

Directrius de desenvolupament d'aplicacions web J2EE

A qui va dirigit

Desenvolupadors d'aplicacions web.

Versió de Canigó

Aquestes directrius no són exclusives del Framework Canigó, ja es tracta d'unes directrius genèriques aplicables a qualsevol aplicació web.

Els **components web de Canigó** suporten la utilització d'**Internet Explorer** (6 i posteriors) i **Firefox**, amb Javascript i Cookies habilitades, com a clients. De totes maneres, qualsevol problema que pugui sorgir a l'utilitzar altres navegadors, us podeu adreçar al Centre de Suport per a obtenir una orientació dels canvis a realitzar.

Introducció

A l'hora de desenvolupar una aplicació web on el canal és Internet, s'ha de tenir en compte que aquesta pàgina podrà ser visitada per múltiples navegadors. Això vol dir que aquesta pàgina ha de ser compatible amb els múltiples navegadors, entenent per compatible que es visualitzi igual o de manera molt similar en els diferents navegadors.

L'objectiu a l'hora de realitzar les proves no és provar-ho amb tots els navegadors existents, però si que és convenient com a mínim fer-ho amb els més importants: Internet Explorer, Firefox, Chrome, Safari, Opera i Mozilla amb les versions més rellevants de cadascun d'ells.

S'ha de tenir en compte que cada navegador renderitza el codi HTML, JavaScript i CSS de manera diferent, entre els navegadors existeixen petites diferències que fan que la visualització final no sigui la mateixa en uns que en d'altres, però també hi ha diferències més importants que poden fer que parts de la pàgina no es vegi o simplement no funcioni.

Per a intentar evitar aquests problemes en aquest document es proposen una sèrie de directrius a seguir a l'hora de desenvolupar una aplicació. Encara que la millor manera és utilitzar els estàndards, l'entitat que regula aquests estàndards és W3C (World Wide Web Consortium).

Consideracions Generals

Com sap un navegador quin estàndard ha de seguir?

Al principi de cada document s'informa una línia amb el **doctype** que informa al navegador quin DTD (Definició de tipus de document) s'està fent servir. Aquest document conté les definicions de l'estructura, elements i atributs d'un llenguatge.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Mode **estricte** vs **peculiar**. Que passa si no s'informa el Doctype d'un document?

Al no indicar un doctype o indicar un doctype incorrecte, el navegador interpreta la pàgina en *Quirks Mode* que es podria traduir com a mode peculiar. Això significa que interpretarà el codi com ho feien

Directrius de desenvolupament d'aplicacions web J2EE

les versions més antigues del navegador en el que s'estigui visualitzant les pàgines i per tant la interpretació diferirà bastant dels estàndards.

En canvi si s'utilitza un doctype vàlid, el navegador interpretarà la pàgina en mode estricte on la interpretació del document serà més ràpida, més similar als estàndards, més consistent i amb un resultat més semblant al que s'espera.

Pots trobar informació detallada en el següent link:

<http://www.w3.org/QA/Tips/Doctype>

Codificació dels documents.

La codificació dels documents, pot variar segons la infraestructura del servidor, però és important informar-la. La codificació per defecte és UTF-8, però es poden utilitzar altres com ISO-8859-1

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

Utilitza una sintaxis correcta.

És recomanable escriure correctament la sintaxi HTML:

- Totes les etiquetes i els seus atributs s'escriuen en **minúscula**.
- Totes les etiquetes s'han de tancar. També les que només utilitzen un tag, com el

- Tots els atributs han d'estar entre cometes dobles "
- Utilitza correctament les etiquetes anidades. S'ha de tenir en compte que hi ha etiquetes que són del tipus block (generen una nova línia) i d'altres inline (en la mateixa línia). Així doncs s'ha d'evitar anidar elements de tipus block dintre d'elements inline.
- És recomanable utilitzar l'atribut title per a incorporar una descripció de l'element, normalment utilitzat en links encara que també pot utilitzar-se en altres etiquetes així com l'alt (text alternatiu) per a les imatges.

No utilitzis etiquetes no estàndards.

Hi ha moltes etiquetes o atributs creades per un navegador i que només funcionen en aquest navegador en especial. Etiquetes com per exemple <marquee>, <blink> atributs com leftmargin entre d'altres que només funcionen en un navegador, no s'han d'utilitzar.

Hi ha d'altres etiquetes que encara que estan suportades per la majoria dels navegadors tampoc s'han d'utilitzar. Algunes d'aquestes són o <center>.

Penseu que la funcionalitat de la majoria d'aquestes etiquetes es poden substituir per fulles d'estil, per tant **és recomanable aplicar estils per el full d'estils i no per etiquetes**. Recorda que els estils han d'anar separats del contingut.

Busca un equilibri en la càrrega dels fitxers estàtics.

És important tenir ben organitzats els fitxers estàtics en diferents fitxers, però pensa que per a cadascun dels fitxers estàtics enllaçats en un document HTML, aquest ha de realitzar una connexió HTTP amb el

Directrius de desenvolupament d'aplicacions web J2EE

servidor i llançar una petició amb les capçaleres corresponents. Tot això pot influir en el temps de descàrrega de la pàgina.

Utilitzo frames?

Cada cop més els *frames* (Marcs) tendeixen a desaparèixer i encara que poden utilitzar-se en una especificació per framesets, no son recomanats. Actualment es recomana l'ús de capes mitjançant l'etiqueta <div>.

Problemes de Cache?

Moltes vegades el navegador no detecta un canvi que hem realitzat en un fitxer estàtic i utilitza una versió *escorcollada* a disc. Impedint que la web mostri els estils correctes o llançant errors javascripts per no existir una funció afegida recentment fins que es força una actualització amb Ctrl + F5.

La solució a aquest problema és molt senzilla, només **cal afegir un paràmetre dinàmic a la crida que es fa al fitxer estàtic:**

```
<link rel="stylesheet" type="text/css" href="/css/estils.css?id=<paràmetre_dinàmic>" />
```

CSS

Fes un reset als estils.

Molts dels valors per defecte que adopten els marges, els tipus de lletra per a les capçaleres <h1>, <h2>, <h3>, ... no son iguals per a tots els navegadors i això pot ser un dels principals problemes a l'hora de que la pàgina es visualitzi correctament en tots els navegadors. Per això **és convenient reiniciar el full d'estils.**

Un dels codis més coneguts per a reiniciar el full d'estils és el *Erik Meyer's Reset*. Aquest codi el podeu trobar a la seva web:

<http://meyerweb.com/eric/thoughts/2007/05/01/reset-reloaded/>

També hi ha altres maneres de realitzar un reset més senzill:

```
* {margin:0; padding:0}
```

Evidentment, aquest reinici de el full d'estils és molt menys complet, però també pot evitar molts problemes.

La nomenclatura.

En les fulles d'estil, fes servir sempre **minúscules** i **utilitza el guió per a separar paraules**. Hi ha navegadors antics que tenen problemes amb les majúscules i minúscules. És important també ser **descriptiu** en els noms i s'ha d'intentar **evitar posar noms que es refereixen a la aparença**.

Utilitza criteris consistents per a organitzar les declaracions d'estils dintre del CSS. Pots afegir les declaracions segons l'ordre d'aparició en el codi o agrupar-les pel tipus d'element al que es refereixen,

Directrius de desenvolupament d'aplicacions web J2EE

però sigui quin sigui el criteri, és important que sigui **constant**. Si no es té una bona organització es pot perdre molt de temps quan les fulles d'estil tenen una mida considerable.

Fulles d'estil per a la impressió.

Es poden carregar diferents fulles d'estil per als diferents medis. Així doncs es poden definir diferents fulles d'estils per a mostrar per pantalla, per imprimir o per a mostrar una pantalla en un dispositiu mòbil. Això és possible mitjançant l'atribut **media** a l'hora de carregar un CSS.

```
<link rel="stylesheet" href="[ruta]/[cssimpressi_o].css" type="text/css" media="print" />
```

Frameworks CSS?

La intenció d'un framework és estalviar temps i minimitzar el risc en el desenvolupament d'aplicacions. Així com existeixen frameworks per a altres llenguatges de programació també existeixen per a les fulles d'estil.

Utilitzar frameworks permet agilitzar el desenvolupament, permet aconseguir que els layouts siguin *cross-browser*, estalviant els problemes habituals entre els diferents navegadors. I permet començar un desenvolupament sobre una base normalitzada.

Però com totes les coses, utilitzar un framework també té els seus riscos: **corba d'aprenentatge**, poden forçar a que utilitzis una **semàntica específica en el codi HTML**, **gran part del codi d'aquest framework mai serà utilitzat**, entre d'altres.

Els frameworks CSS, està bé conèixer-los però no es recomana l'ús. Els més coneguts són: YAML, YUI, Blueprint, ...

JavaScript

La primera cosa que hem de tenir en compte quan utilitzem javascript en la nostra aplicació web, és que el **javascript** és una eina que permet **millorar la usabilitat de les aplicacions**.

S'ha de tenir en compte, que no sempre estarà disponible ja que els diferents navegadors poden deshabilitar-lo, per tant la utilització de JavaScript ha de ser **NO obstructiva**. Algunes de les regles per a utilitzar JavaScript de manera no obstructiva són:

No incloure Javascript directament en la pàgina.

No s'ha d'incloure Javascript directament en la pàgina, cal posar el codi en els fitxers .js i fer referència des de l'html mitjançant la següent línia:

```
<script type="text/javascript" src="scripts.js"></script>
```

Javascript no és un sistema de seguretat.

Pensa que el Javascript es pot deshabilitar i no podem comptar amb ell. Així doncs, les validacions que es facin en javascript, també s'hauran de fer al servidor. Però també s'ha de tenir en compte que la funcionalitat no es pot veure alterada al deshabilitar el javascript. Un exemple seria:

Directrius de desenvolupament d'aplicacions web J2EE

Javascript obstructiu:

```
HTML:
<form action="valida.do">
...
<input type="button" onclick="valida()" value="enviar" />
</form>

Javascript:
function valida() {
  var formulari=document.forms[0];
  //Validacions
...
  if(validacionsOK) {
    formulari.submit();
  }
}
```

Utilitzant javascript d'aquesta manera, si un navegador te deshabilitat el Javascript, no es realitzarà el submit del formulari i per tant aquesta pàgina perdrà tota la seva funcionalitat.

Javascript no obstructiu:

```
HTML:
<form action="valida.do" onsubmit="return valida(this)">
...
<input type="submit" value="enviar" />
</form>

Javascript:
function valida(formulari) {
  //Validacions
  if(validacionsOK) {
    return true;
  } else {
    return false;
  }
}
```

En canvi, si s'utilitza Javascript de manera no obstructiva, encara que el navegador tingui deshabilitat el Javascript, es realitzarà el submit.

Frameworks Javascript?

Així com l'utilització de frameworks CSS no és altament recomanable utilitzar-los, els frameworks Javascript ho poden arribar a ser.

Normalment els frameworks Javascript, son compatibles amb la majoria dels navegadors, així doncs utilitzar un framework t'assegura que aquell script funcionarà correctament en la majoria dels navegadors. Si utilitzes un framework apropiat per al tipus de desenvolupament, faràs les mateixes funcionalitats amb menys codi, t'estalviaràs temps, les funcionalitats que ofereixen aquests frameworks estan optimitzades i normalment una comunitat de desenvolupadors està per darrera.

Directrius de desenvolupament d'aplicacions web J2EE

Alguns dels frameworks javascript més coneguts són:

Mootools → <http://mootools.net/>

jQuery → <http://jquery.com/>

Prototype → <http://www.prototypejs.org/>

YUI → <http://developer.yahoo.com/yui/>

...

Directrius de desenvolupament d'aplicacions web J2EE

Eines

Per sort hi ha eines que ajuden a validar que s'està fent la feina ben feta.

Valida el teu HTML.

<http://validator.w3.org/>

Valida el teu CSS.

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

Comprimeix els fitxers estàtics.

Podeu trobar un HowTo publicat prèviament on s'indiquen aplicacions per a comprimir fitxers estàtics.

<http://canigo.ctti.gencat.cat/confluence/download/attachments/229/Canigó+-+HowTo+-+Eines+de+compressió+JavaScript.pdf?version=1>

Com queda la pàgina en els diferents navegadors?

Existeixen serveis online que permeten veure com es visualitza una pàgina en els diferents navegadors.

Hi ha moltes pàgines d'aquest estil, però una d'elles és:

<http://browsershots.org/>