

Guide to Accessibility

February 2007

 **Generalitat de Catalunya**
gencat.cat



Contents

What is accessibility?	3
How many people are affected by problems of web accessibility?	3
How can we implement accessibility?	3
Accessibility checkpoints	4
Levels of priority	4
Checkpoints and compliance.....	4
Accessibility in Public Administration	4
Web content accessibility guidelines	5
WCAG 1.0.....	5
WCAG 2.0.....	5
WCAG 1.0 Accessibility guidelines	6
Guideline 1 Provide equivalent alternatives to auditory and visual content.	6
Guideline 2 Don't rely on colour alone.....	8
Guideline 3 Use markup and style sheets and do so properly.	9
Guideline 4 Clarify natural language usage.....	12
Guideline 5 Create tables that transform gracefully.....	13
Guideline 6 Ensure that pages featuring new technologies transform gracefully.....	15
Guideline 7 Ensure user control of time-sensitive content changes.	16
Guideline 8 Ensure direct accessibility of embedded user interfaces.....	17
Guideline 9 Design for device-independence.....	18
Guideline 10 Use interim solutions.....	20
Guideline 11 Use W3C technologies and guidelines.	22
Guideline 12 Provide context and orientation information.....	23
Guideline 13 Provide clear navigation mechanisms.	25
Guideline 14 Ensure that documents are clear and simple.....	28
WCAG 2.0 Accessibility guidelines	29
Guideline 1 The content must be perceivable.	29
Guideline 2 The elements in the content interface must be operable.....	30
Guideline 3 The content and the controls must be understandable.....	31
Guideline 4 The content must be sufficiently robust to work with existing and future technologies.....	32
Documentation	33



What is accessibility?

The main purpose of web accessibility is to make the web more easily accessible to all users, independently of circumstances and the devices involved when sources of information are consulted. Based on this idea, an accessible page will be so for a person with a disability or to anyone else whose external circumstances make access difficult (noisy surroundings, situations where visual and auditory information may not be perceived, etc.).

Making a website accessible means understanding that people use the web in different ways. The website must therefore present information in a way which allows people to access it whatever hardware and software they are using and independently of how they navigate round the website. Designers cannot presume that everyone will use IT devices in the same way.

How many people are affected by problems of web accessibility?

One person in every ten in Spain suffers from a disability. About 15% of all the people in Spain with some form of disability live in Catalonia¹. Not all disabilities affect access to IT areas such as internet. For example, difficulty in walking or a coronary defect would not affect access to a website. However, there are many disabilities which do imply a difficulty.

Like any other section of the population, not all those with disabilities have an internet connection. However, the number of internet users is constantly rising and for people with disabilities access to internet is often more vital than it is for the population at large, who have easier access to traditional sources of information, such as printed media.

How can we implement accessibility?

To make web content accessible, the Web Content Accessibility Guidelines 1.0 and Web Content Accessibility Guidelines 2.0 (WCAG 1.0 and WCAG 2.0) have been developed. Their main aim is to guide website design towards an accessible design, thus eliminating barriers to information. In this document we will analyse both sets of guidelines (WCAG 1.0 and WCAG 2.0). These guidelines provide solutions for design and usability, using as examples common situations in which the way a site is designed can make access to information more difficult. The guidelines also include a series of checkpoints which help to detect possible errors.

¹ Data from study carried out by the autonomous communities in 2003 ["Statistical data on discapability in Spain, Europe and Autonomous Communities"](#)



Accessibility checkpoints

Levels of priority

Each checkpoint is assigned one of the three levels of priority established for the guidelines.

Priority level 1: if website developers do not satisfy these checkpoints, certain groups of users will find it **impossible** to access the information on the website

Priority level 2: if website developers do not satisfy these checkpoints, certain groups of users will experience **significant difficulty** in accessing the information on the site.

Priority level 3: if website developers do not satisfy these checkpoints, certain groups of users will experience **some difficulty** in accessing the information on the site.

Checkpoints and compliance

There are three levels of compliance for these checkpoints:

Level "A": all the checkpoints in priority level 1 are complied with. (A)

Level "Double A": all the checkpoints in priority levels 1 and 2 are complied with. (AA)

Level "Triple A": all the checkpoints in priority levels 1, 2 and 3 are complied with. (AAA)

The guidelines describe how to make web pages accessible without sacrificing design, offering the flexibility necessary for the information to be accessible in different situations and providing methods which allow pages to be made useful and intelligible.

Accessibility in Public Administration

In April 2002, referring to the Communication from the "eEurope 2002" Commission, the European Parliament made the following recommendation: "for all websites to be accessible it is fundamental that they comply with the double A (AA) standard and for priority level 2 in the WAI guidelines to be fully implemented".²

In the Spanish context, in July 2002, Law 34/2002 was extended in its fifth additional section: "Accessibility for people with disabilities and for the aged to information provided by electronic media".

This law encourages public administrations to adopt the measures necessary for the information provided on their respective web sites to be accessible for these groups, in accordance with the accessibility criteria agreed in the European Parliament before December 2005.

² http://www.europa.eu.int/information_society/eeurope/index_en.htm



Web content accessibility guidelines

WCAG 1.0

1. Provide equivalent alternatives to auditory and visual content.
2. Don't rely on colour alone.
3. Use markup and style sheets and do so properly.
4. Clarify natural language usage.
5. Create tables that transform gracefully.
6. Ensure that pages featuring new technologies transform gracefully.
7. Ensure user control of time-sensitive content changes.
8. Ensure direct accessibility of embedded user interfaces.
9. Design for device-independence.
10. Use interim solutions.
11. Use W3C technologies and guidelines.
12. Provide context and orientation information.
13. Provide clear navigation mechanisms.
14. Ensure that documents are clear and simple.

WCAG 2.0

1. The content must be perceivable.
2. The elements in the content interface must be operable.
3. The content and the controls must be understandable.
4. The content must be sufficiently robust to work with existing and future technologies.

Indication of the different levels of priority in this document:

Priority level 1: **[P.1]**

Priority level 2: **[P.2]**

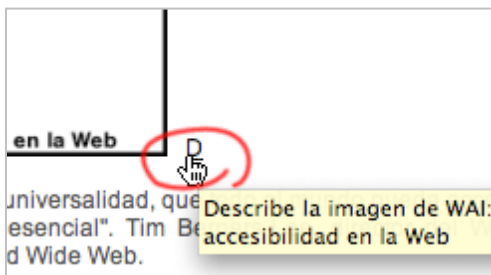
Priority level 3: **[P.3]**

WCAG 1.0 Accessibility guidelines

Guideline 1 Provide equivalent alternatives to auditory and visual content.

Provide equivalent alternatives to visual and audio content which essentially fulfil the same function for the user.

- 1.1. Text equivalent (e.g. Alt + Longdesc) for all non textual elements (images, animations, etc). **[P.1]**
Embedded objects (e.g. presentations en Flash) should contain redundant HTML information so that it is available to all users, independently of the system (browser, plug-ins, versions, etc.) used to connect to the internet.



```
... la Web" longdesc="#imagen_wai">  
... la accesibilidad en la Web">D</a>
```

... mundo pueda acceder a pesar de
... el World Wide Web Consortium e

- 1.2. Provide links in text format for hot spots on an image map. **[P.1]**



```
...ject data="sports.gif" type="im  
<map name="map1">  
<p>  
|<a href="golf.html" shape="re  
  coords="0,0,118,28">Golf</a  
  <a href="motos.html" shape="r  
  coords="118,0,184,28">Moto
```

- 1.3. Provide an audio description of important information in a multimedia presentation.
[P.1]

Use descriptive subtitles for audio events occurring in the presentation.



- 1.4. Provide subtitles, descriptions or similar for all multimedia presentations (e.g. film or animation) **[P.1]**

Use subtitles in multimedia presentations with audio.



Technology for synchronising text with multimedia content:

http://en.wikipedia.org/wiki/Synchronized_Multimedia_Integration_Language

It is also advisable to use transcriptions which, as in the case of embedded objects, allow access to the information independently of the platform being used.

Guideline 2 Don't rely on colour alone.

Make sure that text and graphics are comprehensible whether they are viewed in colour or black and white.

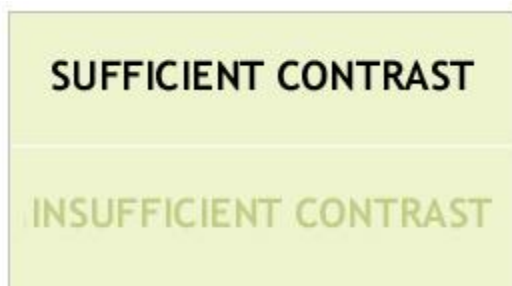
2.1. Use tags (e.g. bold, italic, etc.) to provide information, not colours **[P.1]**

que no pudiera estar conectado a un el
bién acepta los atributos **STYLE, CLAS**
tal como **EM y STRONG** en HTML.
no tiene tal significado. Existe solamen

asi que puede usarse simplemente con
s `EM` y `STRONG</stro`
tal significado. Existe solamente p
uando la hoja de estilo está desactiva

2.2. Provide sufficient contrast between the background colour and the text **[P.2]** and for the images **[P.3]** and the text.

Make sure you provide enough contrast between the background colour and the text to make it easy to distinguish for people with visual defects or problems seeing colour.



A tool is available which allows you to evaluate the level of contrast for different sight problems:

<http://juicystudio.com/services/colourcontrast.php>

Guideline 3 Use markup and style sheets and do so properly.

Create documents with suitable structural elements. Control the presentation with style sheets and not with presentation elements and attributes.

3.1. Use appropriate and valid HTML language. [P.2]

Use HTML tags semantically and in the right place. Don't use images to represent text.

Examples of tags with meaning:

```
title, h1- h6, p, ul, ol, strong, dl
```

Use these tags combined with CSS styles to create the desired effects, so that the presentation and the mark up mean the same:

The following two lines of code could be displayed in the same way by a browser that supports CSS, but if the user uses certain assistive technologies or sees the text without style, the second line conveys meaning while the first does not.

```
<div class="titol">Notícies d'actualitat</div>  
<h2 class="titol">Notícies d'actualitat</h2>
```

3.2. Create documents that are validated by the formal grammars which have been published. [P.2]

Add a declaration of the type of document at the beginning, referring to a published DTD.

e.g.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-  
transitional.dtd">  
<html xml:lang="ca" xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<meta http-equiv="Content-Type" content="text/html;  
charset=iso-8859-1" />  
<title>Generalitat of Catalonia</title>
```

3.3. Use style sheets for layout and presentation. [P.2]

Use the CSS Font property instead of the HTML Font element to control font style.

```
body {  
    margin:0px;  
    padding:0px;  
    font-family: Verdana, Sans-Serif;  
    font-size:101%;  
    color:#000;  
    width: 100%;  
    background-color:#fff; text-align:center;
```

Els treballs de reparació de desperfectes p
de trens. (14.11.2006)

Nova col·lecció d'estudis sobre jo
Els dos nous estudis publicats posen de m
salut dels joves. (14.11.2006)

Ajuts per fomentar l'accés a la so

3.4. Use relative and not absolute measurements in style sheets. [P.2]

Relative units: "em" and "percentage measurements" (%).

Absolute units: "centimetres" (cm), "points" (pt) and "píxels" (px).

Relative measurements allow content to adapt to different browser configurations: font size, screen width, etc. so that the content flows.

3.5. Use header elements to convey document structure and information about the document as specified by W3C. [P.2]

Use "H2" as a subsection of "H1", not to create typographical effects. Headers should follow a logical sequence: there should be no <h3> if we have not used an <h2>.

```
<h1>Generalitat de Catalunya</h1>  
  <h2>Serveis</h2>  
    <ul class="dummy_tab">  
      <li id="tab1">Ciutadania  
      <li><a href='http://www.gencat.cat/>Cultura  
      <li><a href='http://www.gencat.cat/>Esports  
      <li id="tab2">Empreses
```

Generalitat de Catalunya

Serveis

Busqueu l'adreça d'un museu o d'un restaurant populars de Catalunya o descobriu

3.6. Mark up lists and items in lists correctly. [P.2]

Nest elements in "ol", "ul" and "dl" lists properly. Although we can use styles to produce an effect similar to a list, the use of these tags provides additional information about content.

```
<ul class="destacat">  
  <li>  
    <a href="http://mercuri.icc">Mercuri  
  </li>  
  <li>  
    <a href="http://www.gencat.cat">Gencat  
  </li>
```

Arxius

- Ⓧ Els Arxius a Catalunya
- Ⓧ Cercador dels arxius de Catalunya
- Ⓧ Arxiu Nacional de Catalunya

Biblioteques

- Ⓧ Les biblioteques a Catalunya
- Ⓧ Catàlegs col·lectius de la lectura p

- 3.7. Mark quotations. Do not use quotation markers for formatting effects such as indentation. **[P.2]**
In HTML, use “q” and “blockquote” to mark short and long quotations respectively.

```
<blockquote cite="http://www.mycom.  
<p class = "intro">  
    They went in single file, runni  
</p>  
</blockquote>
```

Busqueu l'adreça d'un museu o d'un
populars de Catalunya o descobrir
cultural que teniu més a l'abast.

Els principals tràmits que e
us resulti més còmode.

Seccions

Guideline 4 Clarify natural language usage.

Use markers to facilitate the pronunciation or interpretation of abbreviated or foreign language text.

4.1. Identify language changes in the document. [P.1]

Use the “lang” attribute each time there is a change of language within the text.

e.g.

```
<p> ...la còpia l'haurà controlar el <span lang="eng"> Digital Rights Management</span> amb un...</p>
```

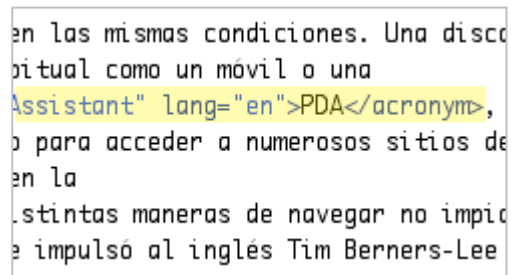
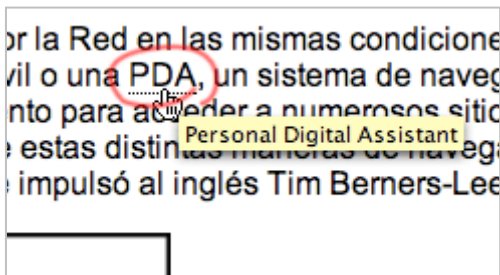
It is advisable to indicate the language in which a page we provide a link to is written if it is different from the language used on our page.

e.g.

```
<a href="www.gencat.net/generalitat/eng/index.htm" hreflang="en">
```

4.2. Identify abbreviations or acronyms with suitable tags when they appear for the first time in the document. [P.3]

Use the tag “abbr” to define abbreviations and “acronym” for acronyms.



4.3. Identify the main language of the document. [P.3]

In the HTML header specify the language which will be used in the document.

e.g.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-  
transitional.dtd">  
<html lang="ca" xmlns="http://www.w3.org/1999/xhtml">  
<head>
```

Guideline 5 Create tables that transform gracefully.

Make sure that tables have the necessary markers to be transformed by accessible browsers and other user applications.

5.1. Identify headers of columns and rows in tables. [P.1]

Use “td” to identify data cells and “th” for headers.

```
<table>
  <tr>
    <th>Nombre</th>
    <th>Tazas</th>
  </tr>
  <tr>
    <td>T. Sexton</td>
```

5.2. Distinguish between header cells and data cells in data tables with two or more levels. [P.1]

Use “thead”, “tfoot”, and “tbody”, to group rows, “col” and “colgroup” to group columns and the “axis”, “scope” and “headers” attributes to describe more complex relationships between data.

```
<table border="1"
  summary="Esta tabla esque
  <caption>Tazas de cafe consum
  <tr>
    <th scope="col">Nombre</
    <th scope="col">Tazas</t
    <th scope="col" abbr="Ti
```

5.3. Don't use tables for layout. [P.2]

Use valid HTML and CSS instead of tables, separating presentation and content.

```
<div id="header">
  
  <ul>
    <li id="t-home"><a href
    <li id="t-note"><a href
```

5.4. Provide summaries of tables. **[P.2]**

Summaries are especially useful for blind users.

```
<table
  summary="Esta tabla esquema
    tazas de cafe con
    el tipo de cafe (
    y si se ha tomad
<tr>
  <th>Nombre</th>
```

5.5. Provide abbreviations for header captions **[P.3]**

Provide short subtitles for header captions, using the “abbr” or “th” attributes. Abbreviations reduce repetition and reading time.

Guideline 6 Ensure that pages featuring new technologies transform gracefully.

Make sure that pages are accessible, even when the latest technology is not supported or is disabled.

- 6.1. Organise the document so that it can be read without style sheets. **[P.1]**
Organise the document in a structured, logical way, using H1, H2, links, contents, etc.



- 6.2. Make sure that pages are still usable when there is no connection, or when scripts, applets, or other programmable objects are not supported. **[P.1]**

e.g.

```
<object classid="java:Press.class" width="500"
height="500">
  <object data="Presion.mpeg" type="video/mpeg">
    <object data="Presion.gif" type="image/gif">
      A medida que la temperatura aumenta, les
      moléculas del globo...
    </object>
  </object>
</object>
```

- 6.3. Provide alternative pages for dynamic content which is not accessible. **[P.1]**
The user should be able to navigate to another page which is accessible, containing the same information, which is updated as often as the inaccessible page.

Guideline 7 Ensure user control of time-sensitive content changes.

Make sure that the user can stop objects or pages which blink, flicker, move or automatically refresh.

7.1. Avoid flicker on the screen. [P.1]

Users with photosensitive epilepsy may suffer attacks caused by flickering or flashing at frequencies between 4 and 59 flashes (hertz) per second, with a peak sensitivity at 20 flashes per second. This also applies to rapid changes from dark to light (as in strobe lights).

7.2. Avoid blinking in content (for example, periodically changing displays and on/off sequences). [P.2]

If you use blinking content (e.g. a header which appears and disappears at regular intervals), provide a means to stop it.

Do not use “blink” or “marquee” elements. These elements do not appear in any W3C specification for HTML.

7.3. Avoid movement on the page. [P.2]

Allow users to stop elements moving.

7.4. Don't create periodically auto-refreshing pages. [P.2]

In HTML, don't create pages which are refreshed with the “http equiv=refresh” code.

```
<meta http-equiv="refresh" content="60">
</head>
<body>

v id="header">
    ”, adding them after each Flash element. These methods make information more readily available to users who do not have Flash, Java, etc. technology available.

e.g.

```
<object
 classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"

 codebase="http://download.macromedia.com/pub/shockwave/cabs/flash
 /swflash.cab#version=4,0,2,0"
 align="baseline" border="0" width="350" height="250">

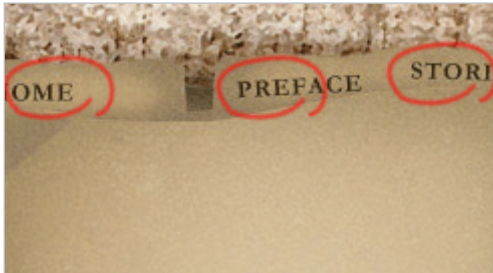
<p align="center">¡MERRY CHRISTMAS AND A HAPPY NEW YEAR!</p>

 <param name="movie" value="navidad.swf">
 <param name="quality" value="high">
</object>
```

## Guideline 9 Design for device-independence.

Use techniques which allow the elements on the page to be activated by various input devices.

- 9.1. Provide client-side image maps instead of server-side image maps, except where the regions cannot be defined with an available geometric shape **[P.1]**



```
a#preface:hover { background-image: url(
 images/prefaceglow.jpg); }
a#stories:hover { background-image: url(
 images/storiesglow.jpg); }
a#gallery:hover { background-image: url(
 images/galleryglow.jpg); }
a#forum:hover { background-image: url(
 images/forumglow.jpg); }
a#mementos:hover { background-image: ur
 images/mementosglow.jpg); }
```

- 9.2. Make sure that any element can be used independently of the device (mouse, keyboard, etc.). **[P.2]**

Make sure it is possible for all users to access the content whether they are using keyboard, mouse, voice input or other peripheral devices. To do this structural HTML elements need to be used (directly accessible for assistive technologies), such as JavaScript, which uses logical event handlers for each situation (and reproduce them for each input device).



9.3. Create a logical tab order through links, form controls, and objects. **[P.2]**

Keep elements in a logical form and order which allows navigation with the keyboard, jumping from one element to another (title, contents, menus, forms, etc.). The order can also be specified by using the “tabindex” attribute.



9.4. Provide keyboard shortcuts for the most important links and for forms. **[P.2]**

Use the “accesskey” tag to provide keyboard shortcuts.

```
<!--links ocults a apartats-->
accés directe a actualitat
accés directe al menú

accés directe al temps

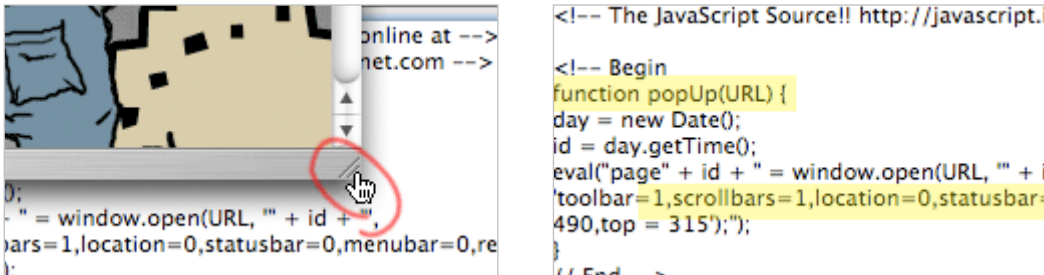
accés a enllaços directes
accés a destacats

```

## Guideline 10 Use interim solutions.

Use provisional accessibility solutions, so that assistive technologies and old browsers work properly.

- 10.1. Make it possible to disable the appearance of new windows. Do not cause new windows to appear suddenly and do not change the current window without informing the user. **[P.2]**



Pop up windows make the user lose control over the content being displayed, which can be disorienting and disconcerting. To improve the user's experience options should be included which allow pop-ups to be blocked.

If a link has necessarily to open a new window or pop-up, an icon should be included to indicate this.

Example of JavaScript to open a new window:

```
<script type="text/javascript"><!--

 function ObrePopUp(e, origen){
 if(!document.captureEvents) e=window.event;

 var boto = (e.which?e.which:e.button);
 var passa = false;

 if(e.type=="click" && (boto==0 || boto==1)) passa = true;
 if(e.type=="keypress" && e.keyCode==13) passa=true;

 if(passa){
 window.open(origen.href, 'popup', '');
 return false;
 }
 }
 if(document.captureEvents) {
 if(Event.KEYPRESS) {
 document.captureEvents(Event.KEYPRESS);
 }
 }
--></script>
```

Example of HTML to open a new window:

```
<a href="http://www.gencat.cat" title="aquest enllaç s'obrirà en
una nova finestra" onclick="return ObrePopUp(event, this)"
onkeypress="return ObrePopUp(event, this)" rel="external">
```

The following URL shows a complete example of an accessible new window:

<http://www.gencat.net/web/test/popup/>

10.2. For all form controls with implicitly associated labels, ensure that the label is properly positioned. **[P.2]**

e.g.

```
en HTML, utilitzeu LABEL i el seu atribut "for".
```

e.g.

```
<label for="nombre">Nombre:
 <input type="text" id="nombre" tabindex="1">
</label>
```

10.3. Include default characters in edit boxes and text areas. **[P.3]**

In HTML, put them in “textarea” and “input”.

e.g.

```
<form action="http://ejemplo.com/prog/text-read"
method="post">
 <p>
 <textarea name=tunombre rows="20" cols="80">
 Teclee su nombre aquí.
 </textarea>
 <input type="submit" value="Enviar"><input
type="reset">
 </p>
</form>
```

## Guideline 11 Use W3C technologies and guidelines.

Use W3C technology (according to specifications) and follow accessibility guidelines. Wherever it is not possible to use W3C technologies or if their use leads to content not being correctly transformed, provide an alternative version of the content which is accessible.

### 11.1. Use W3C technologies when they are available and appropriate for a task and use the latest versions which are supported. [P.2]

See W3C and WAI references for further information.

Validate the syntax of documents: <http://validator.w3.org/>

Validate the syntax of CSS documents: <http://jigsaw.w3.org/css-validator/>

### 11.2. Avoid features not recommended by W3C technologies. [P.2]

In HTML, don't use tags such as "font". In its place use style sheets (CSS), applying the "font" marker in CSS.

Validate documents with automatic accessibility tools. They do not provide all the solutions to accessibility but they detect many common errors.

### 11.3. Provide information so that users can access documents according to their own preferences. [P.3]

Provide information so that users can access documents according to their own preferences, i.e. language, type of content, etc. To achieve this provide links to other versions of the document.

Ex.1:

```
See the French version of this document
```

The language used for the content should also be indicated using HTML markers such as "type" and "hreflang".

Ex. 2:

```

See the French version of this document

```

Ex. 3:

```
See our logo
```

### 11.4. If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page. [P.1]

## Guideline 12 Provide context and orientation information.

Provide context and orientation information to help users to understand the page or any complex elements.

### 12.1. Title each frame to facilitate frame identification and navigation. [P.1]

Use the “title” attribute to describe frame elements.

```
<frameset cols="10%, 90%"
 title="Nuestra bibliotec
 <frame src="nav.html" title
 <frame src="doc.html" title
 <noframes>
 <A href="lib.html" title
 Seleccionar para
 </noframes>
</frameset>
```

The use of “iFrames” is not recommended because they make access more difficult for people with disabilities, since the iFrames content is on another page.

### 12.2. If it is not obvious from the frame title, describe the purpose of the frames and how they are interrelated. [P.2]

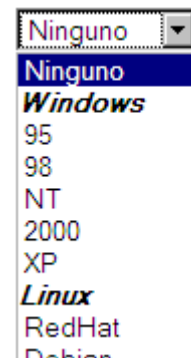
You can use the “longdesc” tag to describe the frame fully.

### 12.3. Divide information into short manageable blocks. [P.2]

Use the “optgroup” tag to group “option” elements in a “select” routine; group form controls with “fieldset” and “legend”; use nested lists when appropriate, use headers to structure documents, etc.

e.g.

```
<select name="so">
 <option selected
value="none">Ninguno</option>
 <optgroup label="Windows">
 <option value="W95">95</option>
 <option value="W98">98</option>
 <option value="WNT">NT</option>
 <option value="W2000">2000</option>
 <option value="WXP">XP</option>
 </optgroup>
 <optgroup label="Mac">
 <option value="MOS9">OS9</option>
 <option value="MOS102">OS 12.2</option>
 <option value="MOS104">OS 10.4</option>
 <option value="MOS106">OS 10.6</option>
 </optgroup>
 <optgroup label="Linux">
 <option value="LRH">RedHat</option>
 <option value="LDeb">Debian</option>
 <option value="LSuSe">SuSe</option>
 <option value="LMDK">Mandrake</option>
```





```
</optgroup>
</select>
```

#### 12.4. Associate labels explicitly with their controls. **[P.2]**

Use the “label” tag and its “for” attribute.

e.g.

```
<label for="nombre">Nombre:
 <input type="text" id="nombre" tabindex="1">
</label>
```

There are some browsers which do not interpret the following version correctly.

```
<label for="identificador">campo</label><input
id="identificador" value="texto" />
```



## Guideline 13 Provide clear navigation mechanisms.

Provide clear and coherent mechanisms for navigation (orientation information, navigation toolbar, website map, etc.) to improve the user's chances of finding what is wanted on the website.

### 13.1. Clearly identify the target of each link. [P.2]

Do not use links of the "see more" variety. Use informative captions.  
Show the target of a link with an informative caption ("title" property of HTML).

If there is no other solution and you have to use texts which are not accessible for links, add a title which clarifies the meaning.

### 13.2. Provide metadata to add semantic information to pages and websites. [P.2]

"Metadata" refers to the information provided about the website and which gives the user orientation information about it.  
There are different kinds of metadata with differing functions.

e.g.

#### title (document title)

```
<title>Welcome to Gencat.cat</title>
```

#### address

```
<address>
Dave Raggett,
Arnaud Le Hors,
personas de contacto del W3C HTML
Activity

$Fecha: 1999/12/24 23:07:14 $
</address>
```

#### meta

#### doctime

#### link

```
<link rel="Next" href="capítulo3">
<link rel="Prev" href="capítulo1">
<link rel="Start" href="portada">
<link rel="Glossary" href="glosario">
```

13.3. Provide information about the general layout of a site. **[P.2]**

For example, site map or contents table, clear mechanisms for navigation, different types of search, metadata and semantic information about the site, etc.  
In the layout description give details of the accessibility characteristics.



13.4. Use navigation mechanisms in a consistent way. **[P.2]**

Create coherent navigation mechanisms. What is expected of an element must be the same throughout the website. It is not good for the same navigation element to behave differently at different points, especially bearing in mind the different types of user who visit a website.



13.5. Group related links, identifying the group. **[P.3]**

Group links and create navigation methods which are intuitive for the user.



- 13.6. If search functions are provided, enable different types of search for different skill levels and preferences. **[P.3]**



The image shows a search interface. At the top, there is a search box with the text 'cercar' and a right-pointing arrow button. Below the search box are two radio buttons: the first is labeled 'a gencat' and is selected, the second is labeled 'al DOGC'. Below this is a section titled 'Directe a' (Direct to) with a dropdown menu showing 'Departaments del Govern' and a right-pointing arrow button. At the bottom of the section are two links: 'Mobilitat' and 'Mapes'.

- 13.7. Place distinguishing information at the beginning of headings, paragraphs, lists, etc. **[P.3]**

Provide information to distinguish different headers, paragraphs and lists.

- 13.8. Provide information about document collections (documents comprising multiple pages.). **[P.3]**

Specify document collections with the “link” element and the “rel” and “rev” attributes. Another way to create a collection is to build a file (e.g. zip, tar or gzip, stuffit, etc.) with the multiple pages.

e.g.

```
<link rel="Next" href="chapter3">
<link rel="Prev" href="chapter1">
<link rel="Start" href="cover">
<link rel="Glossary" href="glossary">
```

- 13.9. Provide a means to skip over multi-line ASCII art. **[P.3]**

You can provide a descriptive equivalent for the ASCII graphic, specifying the data it contains.

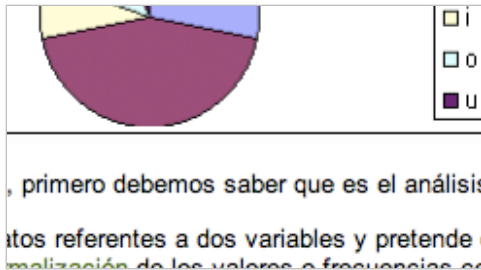
To skip over multi-line ASCII art, provide anchor points to facilitate the process.

## Guideline14 Ensure that documents are clear and simple.

Make sure that documents are clear and simple so that they can be understood more easily.

1. Use the clearest and simplest language appropriate for a site's content. **[P.1]**  
For internet write clearly and concisely in language appropriate for the user. In the following link you can find guidelines and good practice for writing for internet:  
<http://usalo.es/105/escribir-noticias-para-la-web-dos-enfoques/>

2. Supplement text with graphic or auditory presentations where they will facilitate comprehension of the content. **[P.3]**



3. Create a style of presentation that is consistent across pages. **[P.3]**

<b>categoría «Centrado al Usuario»</b>		categoría Centrado al
		buscabas genial, sino
		tienes el buscador y
		y el archivo tempora
<a href="#">Nintendo Wii:</a>	<a href="#">Diseño, Centrado al</a>	11 de Mayo de 2006
<a href="#">Shigeru Miyamoto</a>	<a href="#">Usuario, Juegos de</a>	
<a href="#">jugando al tenis,</a>	<a href="#">Consola, Juegos</a>	



## WCAG 2.0 Accessibility guidelines

### **Guideline 1** The content must be perceivable.

1. Text equivalents should be provided for any non-text content, with the same purpose and providing the same information, except when the non-text purpose is to provide a specific sense experience (e.g. music or visual art). In such cases a label or text description is enough.  
See Ref. Point 1.1 - Guideline 1 in WCAG 1.0
2. In time-sensitive presentations, provide information by other synchronised means.  
See Ref. Point 1.3 - Guideline 1 in WCAG 1.0
3. Check that information, functionality and structure are independent of presentation.  
See Ref. Point 3.3 - Guideline 3 in WCAG 1.0
4. In presentations which are visual by default, check that the foreground content (words and/or images) can be distinguished easily from the background.  
See Ref. Point 2.2 - Guideline 2 in WCAG 1.0
5. In presentations which use sound by default, check that the main content (speech and other sounds) can be easily differentiated from background noise. **[P.2]**  
See Ref. Point 1.3 - Guideline 1 in WCAG 1.0

---

The aim is to produce robust documents, separating content (information and structure), presentation (CSS styles) and behaviour (JavaScript), to facilitate reading for different browsers, and to make document maintenance easier: error correction, adapting to new standards, development, etc.

To do this we need to produce documents which provide as much information as possible, even if they have non-HTML content (images, embedded objects, Flash, audio clips, etc.), offering alternatives to the information they contain.



## **Guideline 2** The elements in the content interface must be operable.

1. All the functionalities of the content must be designed so that they are operable by means of a keyboard interface.  
See Ref. Point 9.2 - Guideline 9 in WCAG 1.0
  2. Allow users to control any time limit on reading, interacting or responding to a document, except when this is not possible because of the nature of the content or because of actions occurring in real time.  
See Ref. Point 7.3 - Guideline 7 in WCAG 1.0
  3. Allow users to avoid content which could cause photo-sensitive epileptic seizures.  
See Ref. Point 7.1 - Guideline 7 in WCAG 1.0
  4. Give users the means to orientate themselves and navigate through the content.  
**[P. 2]**  
See Ref. Points 9.1, 9.2, 9.3 and 9.4 - Guideline 9 in WCAG 1.0
  5. Help users to avoid errors and correct them easily. **[P. 2]**  
See Ref. Point 8.1 - Guideline 8 in WCAG 1.0
- 

It is important to programme content which is independent of access systems (browser, assistive technology, etc.) and the devices we use to interact with a website (keyboard, mouse, etc.).

When technology for communicating with the server, such as AJAX, is used, we must keep the user informed at all times. These technologies load certain webpage content without modifying the rest, which the user should be aware of. Similarly, when there is an error, the information should be visible and clear to the user, providing a suitable alternative.

Users must be in control of the context in which they find themselves: they must be able to enable or disable functionalities such as time limits, alert messages, etc.



### **Guideline 3** The content and the controls must be understandable.

1. Make sure that the meaning of the content can be perceived automatically.  
See Ref. Points 14.1, 14.2 and 14.3 - Guideline 14 in WCAG 1.0
  2. Organise content coherently page by page and make sure interactive elements behave predictably.  
See Ref. Guidelines 12, 13 and 14 in WCAG 1.0
- 

It is important to identify the language used in each document, as well as any change in language (e.g. a sentence in French within a document in English), so that voice synthesisers can use the correct pronunciation and make the content easier to understand, so that dictionaries can be accessed, etc.

Provide the meaning of abbreviations and acronyms to make them more intelligible.

Defining key words and specialised vocabulary will help to produce a more robust document.

Writing simply and clearly will benefit not only users with **cognitive, learning or reading disabilities** but also those from other countries who speak other languages or those who use sign language to communicate. This does not exclude expressing complex or technical ideas.

Sound, graphics, video and animation can help users to understand the concepts which appear in the website, especially people with cognitive, learning or reading disabilities or those who are not familiar with the language used in the text on the website.

Summaries of information which may be difficult to understand help those who have difficulty reading.

Summaries of visual content, which explain the meaning of complex information, help people who are blind, suffer from dyslexia or have sight problems to understand information better.



**Guideline 4** The content must be sufficiently robust to work with existing and future technologies.

1. Use technologies according to specifications.  
See Ref. Point 6.1 - Guideline 6 in WCAG 1.0  
See Ref. Points 11.1 and 11.2 - Guideline 11 in WCAG 1.0
2. Make sure that user interfaces are accessible or that accessible alternatives are provided.  
See Ref. Points 1.1, 1.2 and 1.4 - Guideline 1 in WCAG 1.0  
See Ref. Point 6.1 - Guideline 6 in WCAG 1.0

---

Using standards guarantees forward compatibility (with more modern technologies) and to a great extent backward compatibility (with older technologies). HTML mark-ups which are specific to certain browsers (blink, marquee, etc.) should be avoided, as should CSS properties not defined by W3C and JavaScript which is browser specific. If we use JavaScript which is personalised for certain browsers, we make maintenance more complicated as well as forward compatibility (to new technologies).



## Documentation

Below we provide a series of links to fuller and more extensive documentation than that provided in this guide.

If you have any questions or suggestions concerning this guide, please submit them via the e-Catalunya platform <http://ecatalunya.gencat.net> to the Internet group in the Citizen attention portal, if you are a registered user. Otherwise, you can write to [suport@gencat.net](mailto:suport@gencat.net).

If you need clarification about W3C accessibility guidelines, you can contact the consortium (the addresses of the offices in Spain and the e-mails of people to contact can be found in the [organisation's website](#)).

### **W3C - World Wide Web Consortium**

Internet regulatory body.

<http://www.w3c.es/>

### **WAI - Web Accessibility Initiative**

Regulatory body for web accessibility, within W3C.

<http://www.w3c.es/Traducciones/es/WAI/intro/accessibility>

### **WCAG 1.0 – Web Content Accessibility Guidelines 1.0**

Web content accessibility guidelines, as in the 5th May 1999 version.

<http://www.w3.org/TR/WAI-WEBCONTENT/>

### **WCAG 2.0 – Web Content Accessibility Guidelines 2.0**

Web content accessibility guidelines, updated 19th November 2004.

<http://www.w3.org/TR/WCAG20/>

### **Official translation into Catalan – Web content accessibility guidelines 1.0**

Web content accessibility guidelines, as in the 5th May 1999 version.

[http://www.w3.org/2006/11/WCAG\\_ca/](http://www.w3.org/2006/11/WCAG_ca/)

### **HTML techniques for web accessibility 1.0**

Guidelines and techniques for creating and adapting HTML content.

<http://www.w3.org/TR/WCAG10-HTML-TECHS/>

### **CSS techniques for web accessibility 1.0**

Guidelines and techniques for creating and adapting CSS content.

<http://www.w3.org/TR/WCAG10-CSS-TECHS/>

### **List of validation guidelines for WCAG 1.0**

Appendix to WCAG 1.0 document, organised by concepts.

<http://www.w3.org/TR/WCAG10/full-checklist.html>

### **Validation of document syntax**

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

### **Validation of CSS document syntax**

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