

Software y estándares para la Web

P0. ENTORNO

Contenido

Objetivos	2
Comprobación del puesto de trabajo	2
Editores	2
Brackets.....	2
Editor NotePad++	3
Navegadores.....	3
Chrome	3
Firefox.....	4
Microsoft Edge	5
Opera.....	5
Safari (iOS).....	6
Cliente SSH-SFTP – Bitvise	7
Conexión a la máquina RITCHIE	7
Creación de <i>public_html</i>	9
Archivo <i>index.html</i>	9
Validadores.....	10
HTML https://validator.w3.org/	10
CSS https://jigsaw.w3.org/css-validator/	12
Accesibilidad Web	16
Herramienta TAW - http://www.tawdis.net/	16
Herramienta Wave - http://wave.webaim.org/	18
Herramienta aChecker - https://achecker.ca	19
Test de adaptabilidad (<i>responsive</i>)	20
Google Mobile-friendly – https://search.google.com/test/mobile-friendly	20
Screenfly - http://quirktools.com/screenfly/	21
Anexo I. Recordatorio de permisos en sistemas UNIX.....	22

NOTA: Si se copian los códigos HTML y CSS directamente de este archivo mediante Ctrl+C y Ctrl+V en el editor, se introducen caracteres “extraños”.

Para evitar los inconvenientes, se proporciona el archivo “*P0.Recursos.zip*” con los códigos identificados con la etiqueta en cada apartado de este guion.

Html1

Objetivos

En esta práctica se va a realizar la comprobación del entorno de trabajo del estudiante además de la configuración de la cuenta personal de los alumnos en la máquina Ritchie para poder alojar un sitio Web. Se realizará este sitio web utilizando los estándares que se explicarán en la asignatura. Se comprobará el cumplimiento de los estándares y el nivel de accesibilidad.

Comprobación del puesto de trabajo

Las herramientas que vamos a utilizar en la asignatura son:

- Editores: Para editar el código de los estándares y lenguajes vamos a utilizar fundamentalmente dos editores, Brackets y NotePad++
- Navegadores: Para visualizar los sitios web. Se utilizarán los navegadores más comerciales en la actualidad, Chrome, Firefox, Microsoft Edge y Opera. Para sistemas OSX e iOS Safari.
- Cliente SSH-SFTP: para conectarse a Ritchie y a otras máquinas, Bitvise

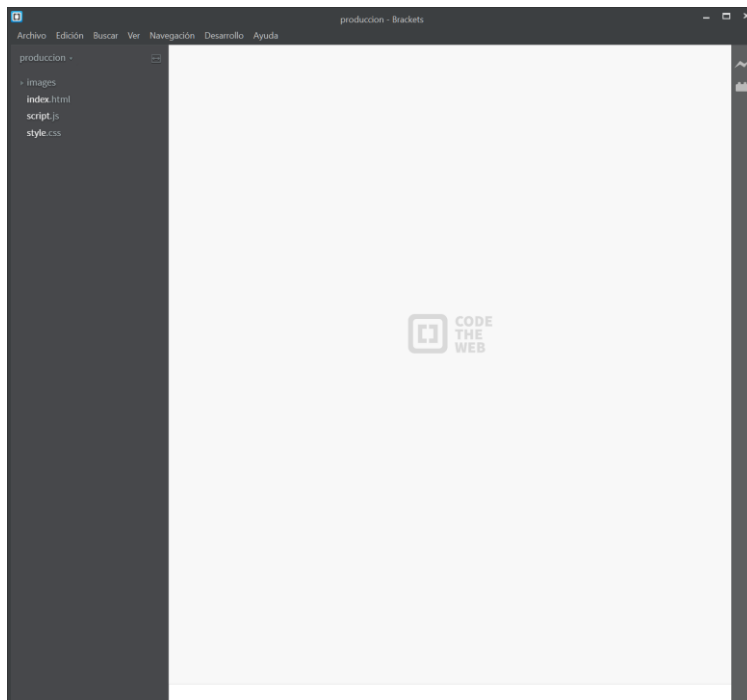
Editores



Brackets

Para editar el código de los estándares y lenguajes vamos a utilizar el editor Brackets (<http://brackets.io/>), es un editor open-source escrito en HTML, CSS y JavaScript creado por Adobe Systems bajo licencia MIT License y mantenido a través de GitHub. Tiene versiones para Mac, Windows y Linux.

Brackets utiliza codificación en UTF-8.

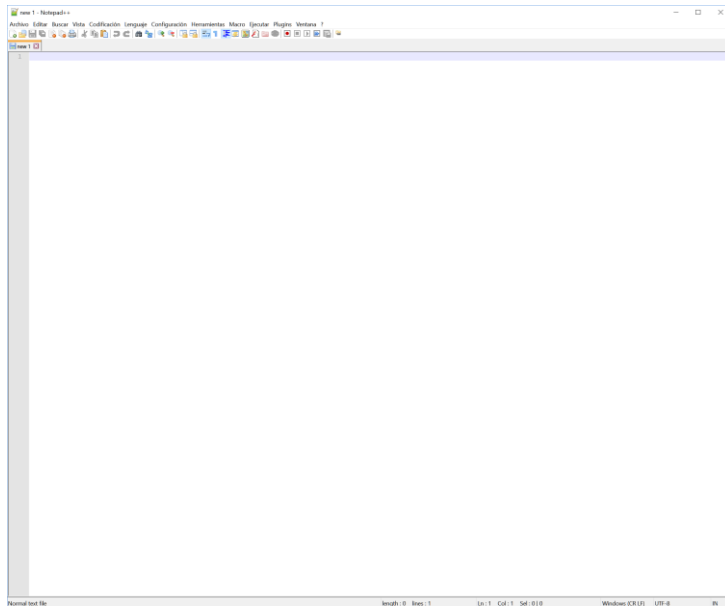




Editor Notepad++

Editor de código distribuido como software libre, hospedado inicialmente en SourceForge.net y desde 2015 en GitHub.

NotePad++ utiliza diversas codificaciones de los archivos, en el caso de esta asignatura se debe seleccionar codificación UTF8 sin BOM.



Navegadores

Vamos a utilizar varios navegadores en la asignatura, siempre deberemos comprobar la visualización de los sitios web en todos ellos.

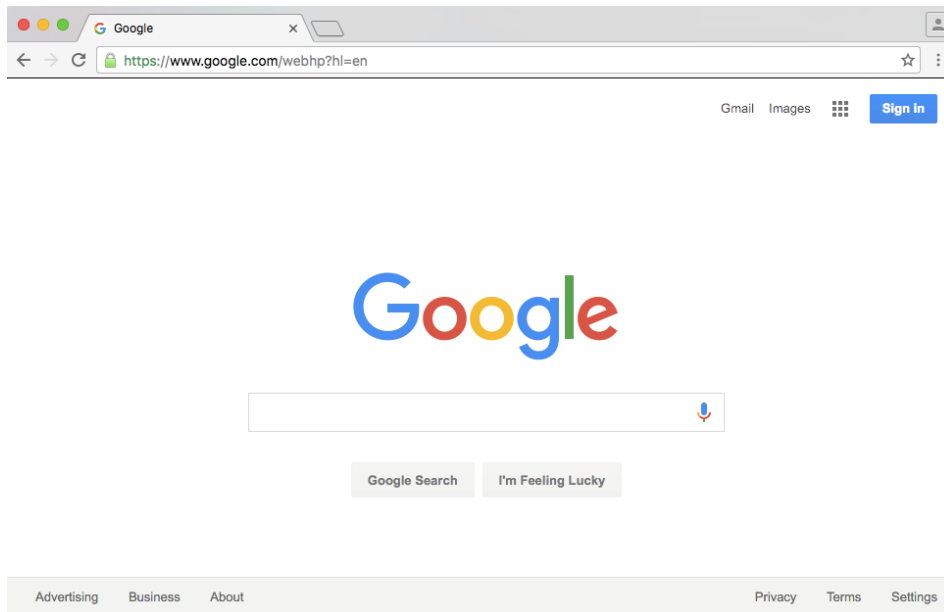


Navegador web desarrollado por Google y compilado con base en varios componentes e infraestructuras de desarrollo de aplicaciones (frameworks) de código abierto.

Lenguaje De Programación: C++, Ensamblador, Python, JavaScript

Motor de renderizado Blink (fork de WebKit)

Fecha del lanzamiento Inicial: 2 de septiembre de 2008

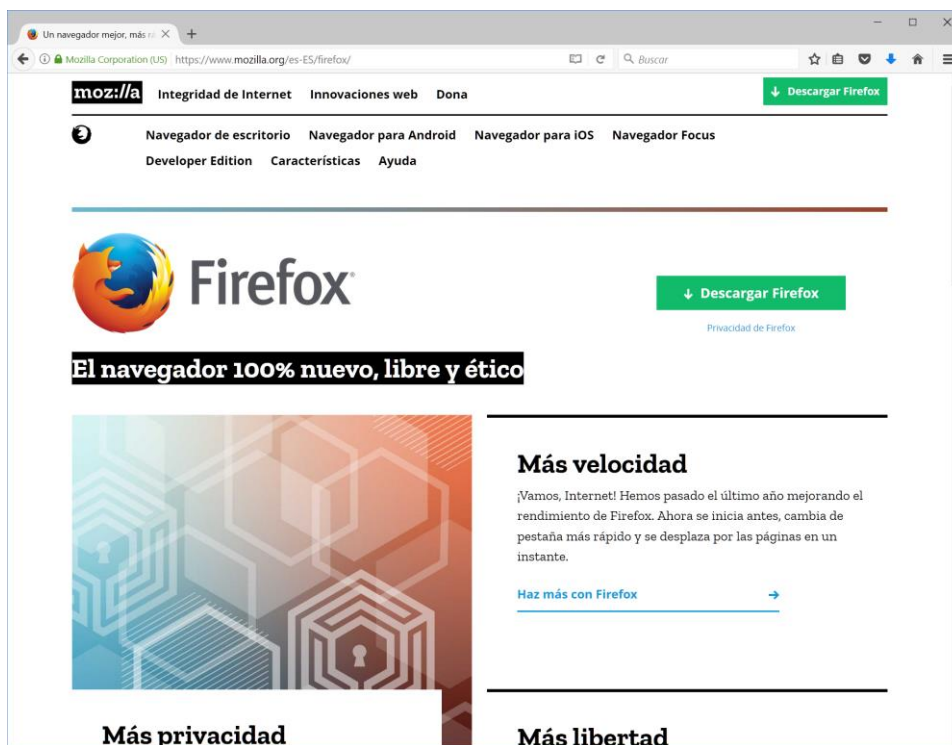


Mozilla Firefox es un navegador web libre y de código abierto desarrollado para Linux, Android, IOS, OS X y Microsoft Windows coordinado por la Corporación Mozilla y la Fundación Mozilla. Usa el motor Gecko para renderizar páginas web, el cual implementa actuales y futuros estándares web.

Lenguaje De Programación: C++, XUL, XBL, JavaScript, CSS2

Motor de renderizado Gecko

Fecha del lanzamiento Inicial: 23 de septiembre de 2002



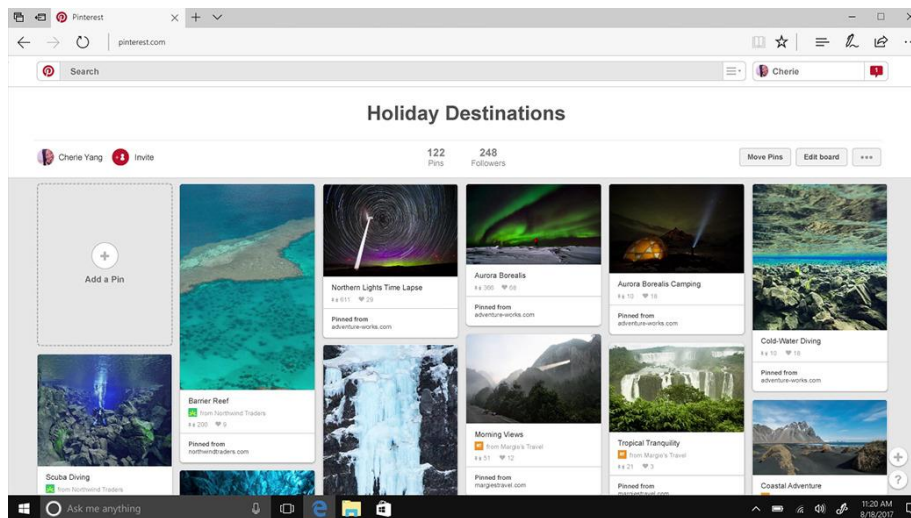
Microsoft Edge

Navegador web desarrollado por Microsoft, que se encuentra incluido en Windows 10, está diseñado para ser un navegador web ligero con un motor de renderizado de código abierto construido en torno a los estándares web, y Microsoft se ha comprometido a actualizar Edge para integrarlo con los nuevos y existentes estándares que aún no soporta.

Lenguaje De Programación en C++

Motor de renderizado EdgeHTML que deriva de Trident

Fecha del lanzamiento Inicial: 29 de julio de 2015



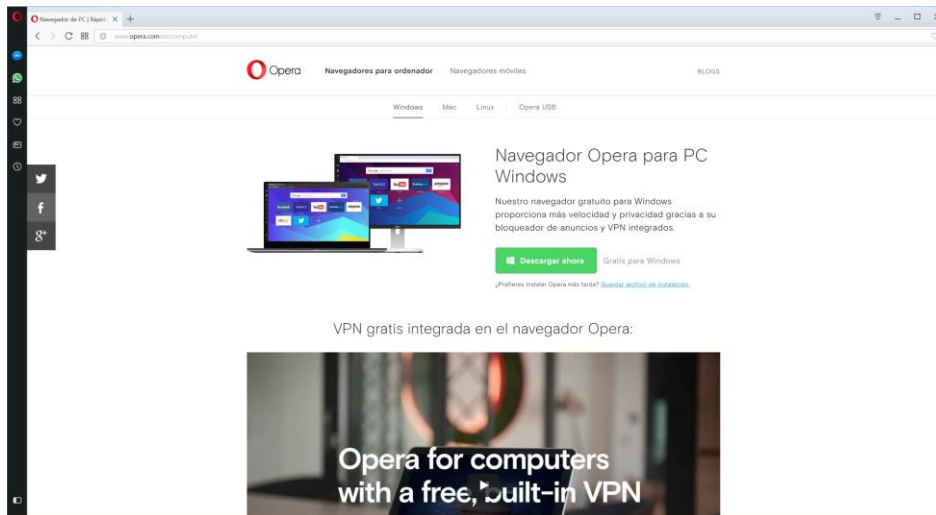
Opera

Opera es un navegador web creado por la empresa noruega Opera Software. Usa el motor de renderizado Blink. Tiene versiones para ordenadores de escritorio, teléfonos móviles y tabletas.

Lenguaje De Programación: C++

Motor de renderizado Blink

Fecha del lanzamiento Inicial: 10 de abril de 1995



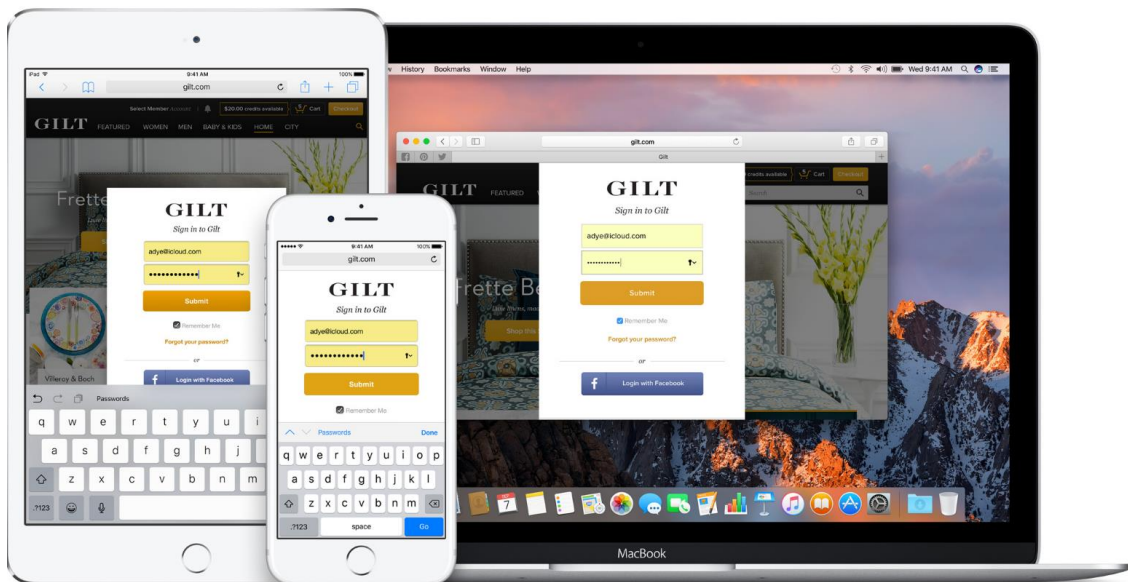
Safari (iOS)

Safari es un navegador web de código cerrado desarrollado por Apple Inc. Está disponible para macOS, iOS.

Lenguaje De Programación: C++, Objective-C y Swift

Motor de renderizado basado en Webkit

Fecha del lanzamiento inicial: 7 de enero de 2003





Ciente SSH-SFTP – Bitvise

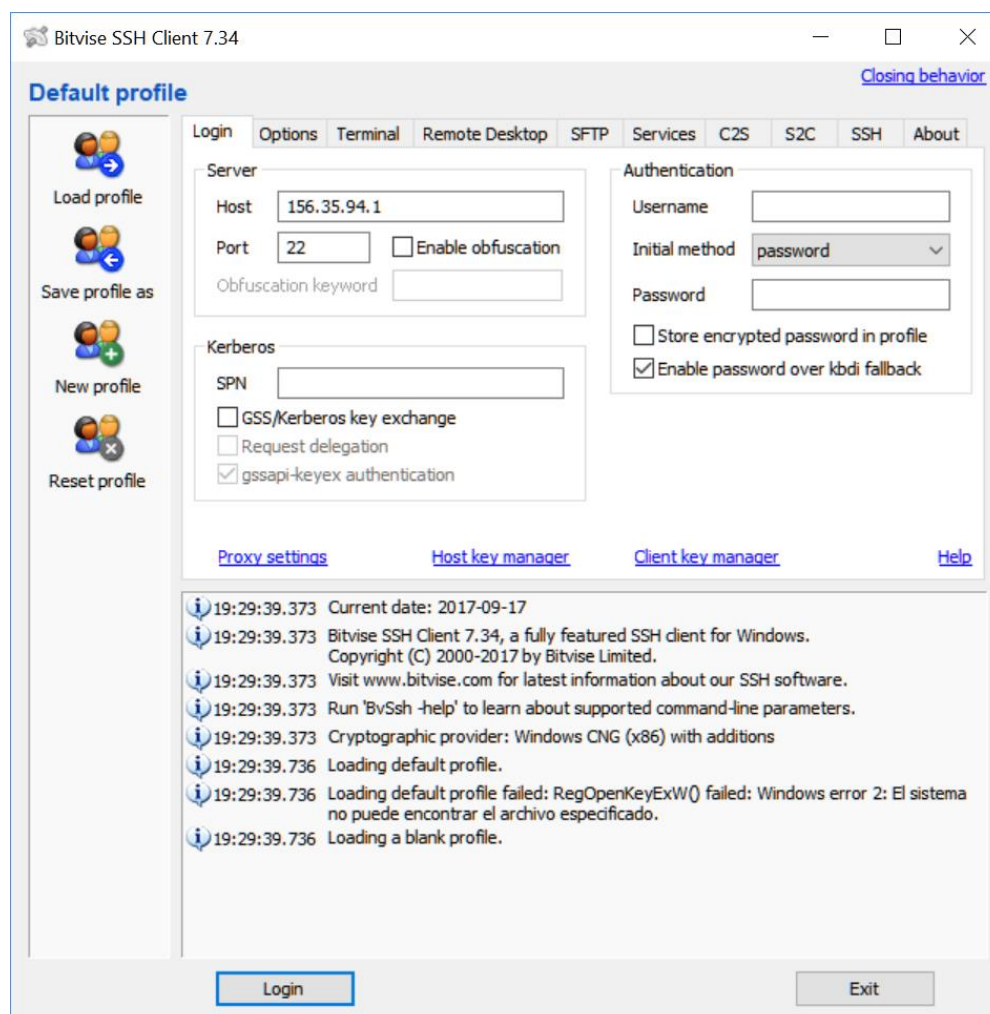
Para conectarnos a Ritchie necesitamos una suite de herramientas que utilicen el protocolo SSH para conectarnos de forma remota y segura. En concreto Bitvise.

Conexión a la máquina RITCHIE

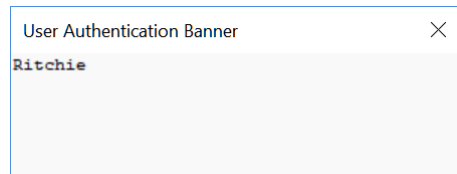
La Escuela de Ingeniería Informática proporciona una cuenta en el servidor RITCHIE para cada estudiante donde pueden realizar sus prácticas.

Ejecutamos el Ciente SSH Bitvise y se muestra la ventana donde se debe especificar la máquina y las credenciales.

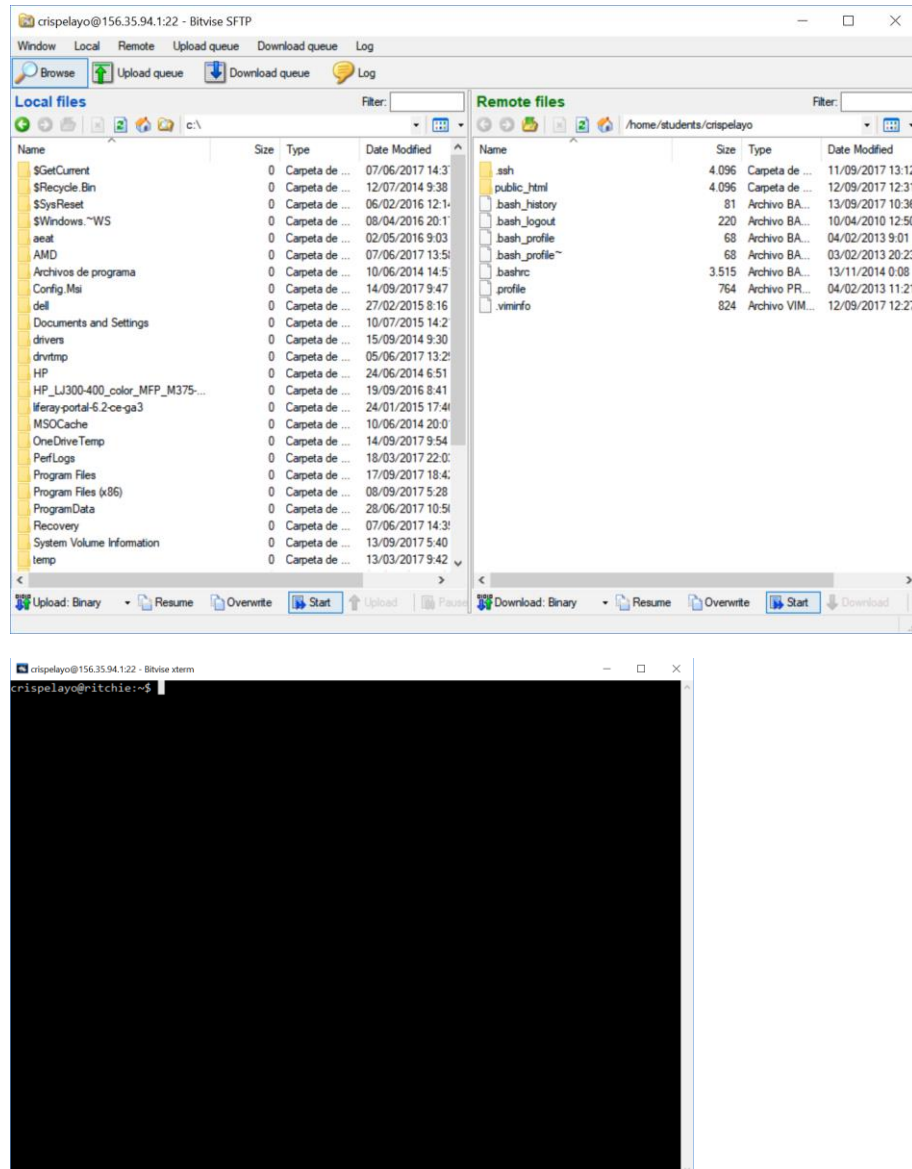
- IP de RITCHIE 156.35.94.1
- Usuario: UOxxxxxx (cada estudiante su UO)
- Password: la proporcionada



Si se ha conectado correctamente aparece la siguiente ventana que debe ser cerrada



En ese momento ya aparece la ventana de SFTP y la terminal



Creación de *public_html*

Vamos a realizar la configuración de esta cuenta personal para que albergue la página de la asignatura.

- Accedemos a RITCHIE (156.35.94.1)
- Comprobamos el contenido y los permisos

```
$ls -al
```
- Si no existe el directorio *public_html* lo crearemos

```
$mkdir public_html
```
- Comprobamos los permisos de *public_html*. Los permisos mínimos deben ser (véase Anexo I):
 - Propietario: acceso total (lectura, escritura y ejecución)
 - Grupo: ninguno
 - Resto: ejecución

```
$chmod 701 public_html
```

Archivo *index.html*

El archivo *index.html* es el punto de entrada de la página web. Cuando en un navegador Web se escribe la dirección de una página web busca este fichero y lo muestra.

Vamos a realizar la creación de este archivo:

- Abrir el editor de texto Brackets (si se utiliza Notepad++ especificar codificación UTF8 sin BOM)
- Crear un nuevo archivo y copiar el siguiente código

```
<!DOCTYPE HTML>
<html lang="es">
<head>
  <!-- Datos que describen el documento -->
  <meta charset="UTF-8">
  <title>Comenzando</title>
</head>

<body>
  <!-- Datos con el contenido que aparece en el navegador -->
  <h1>¡Hola mundo!</h1>
</body>
</html>
```

Html1

- Guardar el archivo con nombre *index.html*
- Abrir el cliente Bitvise
- Acceder a RITCHIE y subir el fichero *index.html* al directorio *public_html*

- Desde el cliente SSH modificar los permisos del archivo. Los permisos deben ser de lectura y escritura para el propietario, ninguno para el grupo y de ejecución para el resto de los usuarios.
`$chmod 604 index.html`
- Probar en los navegadores Web que se accede a la Web y aparece una página del saludo.
- URL: `http://156.35.94.1/~UOxxxxxx`

Validadores

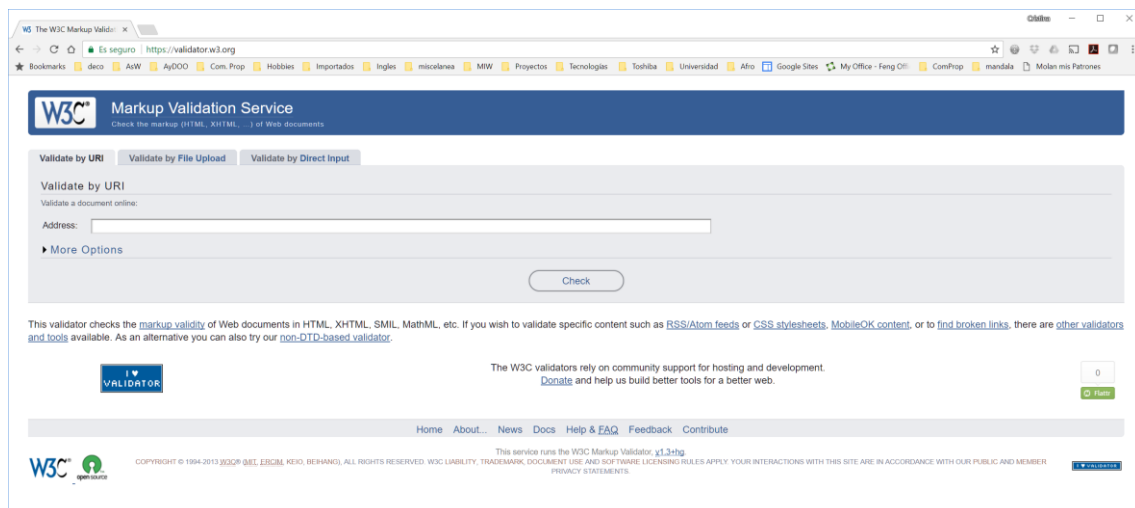
HTML <https://validator.w3.org/>

Los documentos HTML se inician con elementos que indican a los navegadores cómo deben interpretarlos y a los servicios de validación cómo deben probarlos.

El W3C proporciona un servicio online y gratuito que nos permite comprobar la validez de los documentos HTML. Más información https://www.w3.org/wiki/Validating_your_HTML

Pasos:

- Abrir en el navegador la página <http://validator.w3.org/>



- Seleccionar la forma de validación:
 - Por URL: introducir la dirección de RITCHIE `http://156.35.94.1/~UOxxxxxx`
 - Por fichero: seleccionando el fichero local
 - Copiando directamente el código
- Pulsar en “Check” que corresponda

El resultado de la validación puede ser:

- HTML Válido

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for <http://156.35.94.1/~crispelayo/>

Checker Input

Show

☐ source ☐ outline ☐ image report

Options...

Check by

address

<http://156.35.94.1/~crispelayo/>

Check

Document checking completed. No errors or warnings to show.

Used the HTML parser.
Total execution time 385 milliseconds.

- HTML Inválido: se muestran los errores que deben ser revisados y corregidos

Para forzar un error vamos a eliminar la etiqueta de cierre `</h1>` del encabezado de la página. Pasamos el validador y el resultado será:

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show

☒ source ☐ outline ☐ image report

Options...

Check by

text input

```
<!DOCTYPE HTML>

<html lang="es">
<head>
  <!-- Datos que describen el documento -->
  <meta charset="UTF-8">
  <title>Comenzando</title>
</head>

<body>
  <!-- Datos con el contenidos que aparece en el navegador -->
  <h1>¡Hola mundo!
</body>
</html>
```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

1. **Error** End tag for `body` seen, but there were unclosed elements.

From line 13, column 1: to line 13, column 7

```
a mundo! <--></body><--></html>
```

2. **Error** Unclosed element `h1`.

From line 12, column 5: to line 12, column 8

```
<!--><h1>¡Hola
```

Los errores suelen ser descriptivos, y en cualquier caso siempre señalan la línea donde se ha producido el error para su localización más fácil.

Una vez que el documento es HTML Válido podemos añadir a nuestro documento el icono de conformidad HTML5.

Página 11 de 23

Html2

```
<footer>

  <a href="https://validator.w3.org/check?uri=referer">

  </a>

</footer>
```

Lista completa de iconos de Validación del W3C <https://www.w3.org/QA/Tools/Icons> y el logo de HTML5 <https://www.w3.org/html/logo/>



CSS <https://jigsaw.w3.org/css-validator/>

Las Hojas de Estilo en Cascada (Cascading Style Sheets), son el mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. Página Web de CSS: <http://www.w3.org/Style/CSS/>

Vamos a crear una hoja de estilo muy básica para comprobar el funcionamiento del validador.

- Crear en el editor de texto un archivo denominado “estilo.css”
- Añadir el siguiente código:

Css1

```
body {
  font: 400 15px Tahoma; /*Familia sans-serif*/
  line-height: 1.8;
  color: #818181;
}
h1 {
  background-color:#FF5733;
  color: #fff;
  text-transform: uppercase;
  padding: 100px 25px;
  text-align: center;
}
```

- Editar el fichero “index.html”
- Añadir el enlace a la hoja de estilo, en la cabecera detrás de <title>:

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

- Subir a RITCHIE los ficheros dentro de “public_html”
- Probar en los navegadores el nuevo aspecto de la página

Vamos a realizar la validación de la hoja de estilo que hemos creado.

Pasos:

- Abrimos el validador <https://jigsaw.w3.org/css-validator>

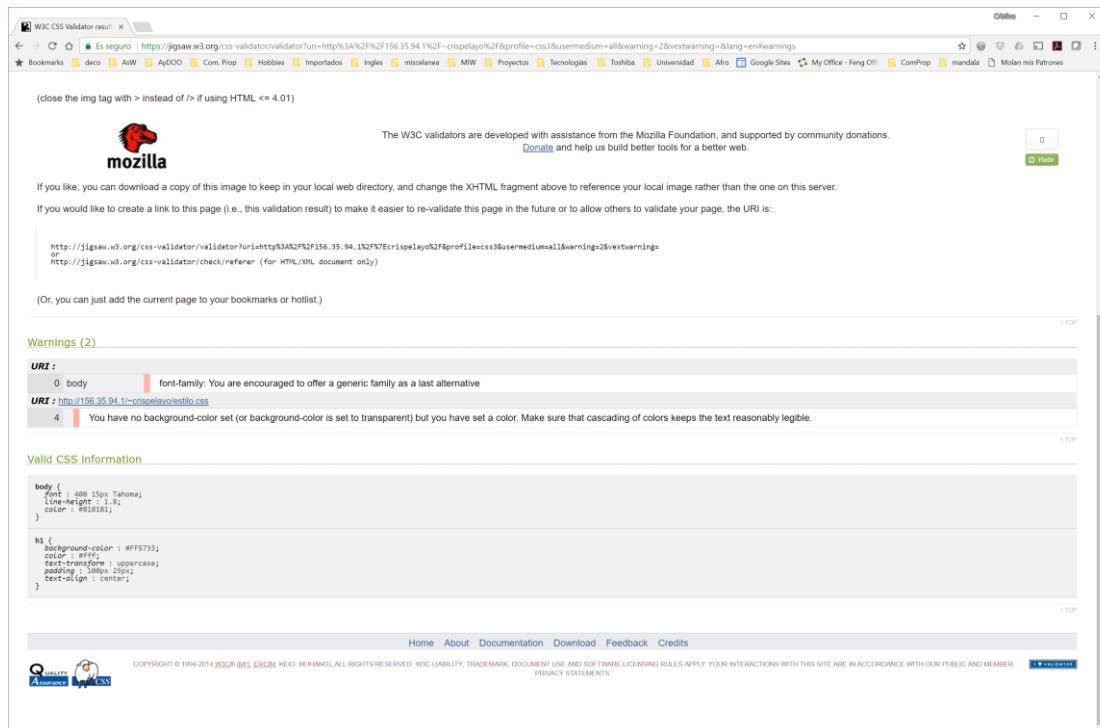
- Seleccionar la forma de validación:
 - Por URL: introducir la dirección de RITCHIE <http://156.35.94.1/~UOxxxxxx>
 - Por fichero: seleccionando el fichero local
 - Copiando directamente el código
- Pulsar en “Check”

El resultado de la validación puede ser:

- CSS válido

- CSS válido con warnings que deben ser resueltos. Cambiamos en el validador en “More Options” y la opción “Warnings = All” para que nos proporcione un listado completo de

los warnings de la hoja de estilo. Volvemos a validar y comprobamos que tenemos 2 warnings.



Para solucionar los warnings debemos añadir la familia genérica de la fuente definida para h1 y el background-color dentro de body.

```
body {  
  font: 400 15px Tahoma, sans-serif;  
  line-height: 1.8;  
  color: #818181;  
  background-color: #f6f6f6;  
}  
h1 {  
  background-color: #FF5733;  
  color: #fff;  
  text-transform: uppercase;  
  padding: 100px 25px;  
  text-align: center;  
}
```

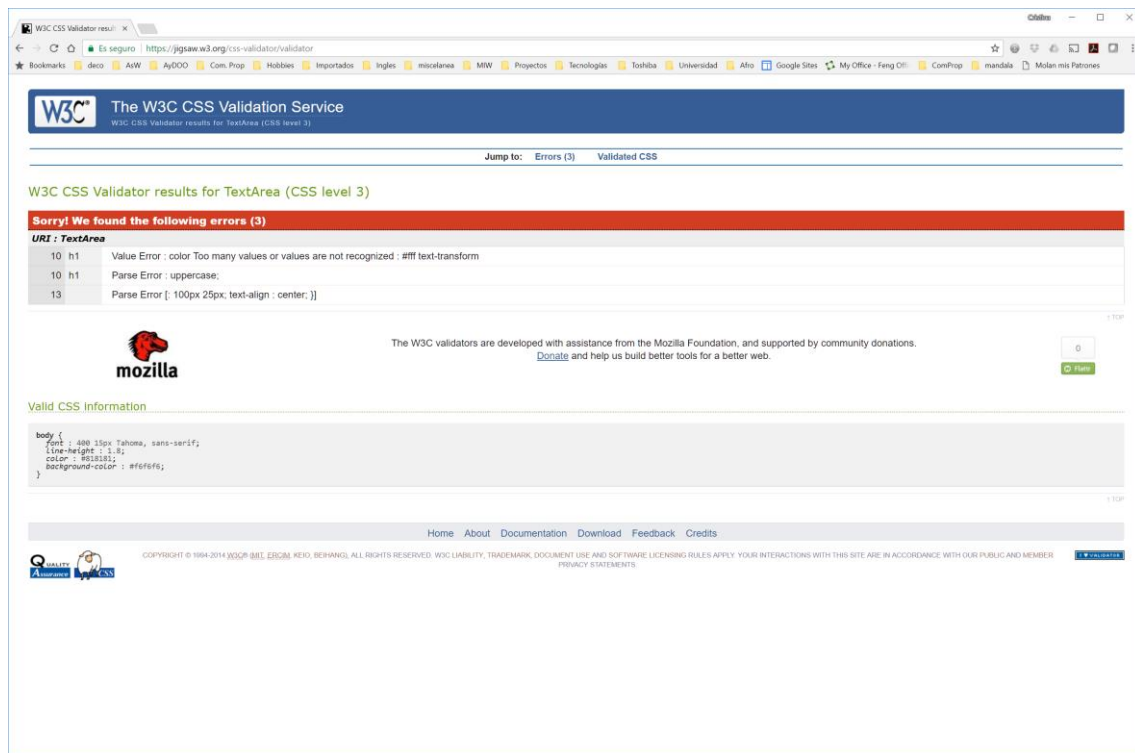
Css2

- CSS inválido

Para forzar el error eliminamos el “,” de la regla color del encabezado h1

```
color: #fff
```

Pasamos el validador y el resultado será:



Los errores se detallan en cascada en los navegadores, siempre se debe solucionar el primero, validar y comprobar si se han solucionado el resto.

Una vez que el documento es CSS Válido podemos añadir a nuestro documento el icono de conformidad CSS.

Html3

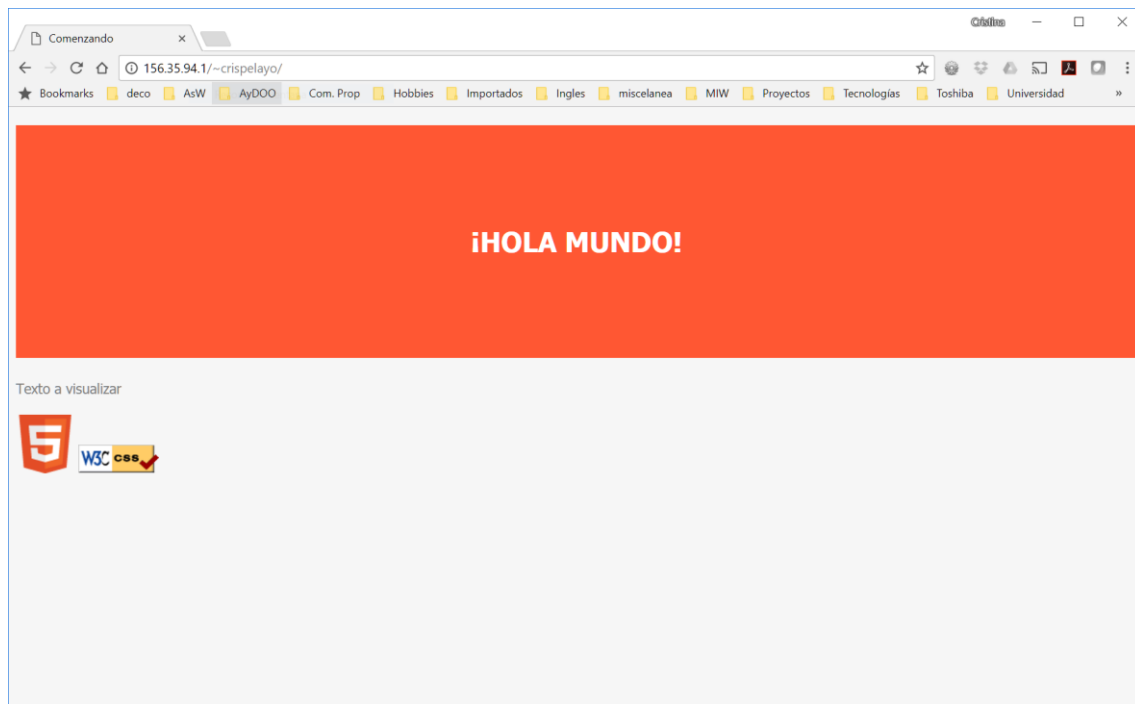
`<footer>`

```
<a href="http://jigsaw.w3.org/css-validator/check/referer ">
  </a>
```

`</footer>`

Se utiliza el icono oficial que propone el validador del W3C pero no es obligatorio utilizar ese.

El resultado visual de la página realizada es:



Accesibilidad Web

La Accesibilidad Web trata del acceso universal a la Web, independientemente del tipo de hardware, software, infraestructura de red, idioma, cultura, localización geográfica y capacidades de los usuarios.

El W3C desarrolla la “Iniciativa de Accesibilidad Web”, conocida como WAI (Web Accessibility Initiative) su objetivo es facilitar el acceso de las personas con discapacidad, desarrollando pautas de accesibilidad, mejorando las herramientas para la evaluación y reparación de accesibilidad Web.

Las herramientas que vamos a utilizar en la asignatura proporcionan un análisis automático de la accesibilidad que siempre debe completarse con un análisis manual de las pautas.

Herramienta TAW - <http://www.tawdis.net/>

TAW es una familia de herramienta para el análisis de la accesibilidad de sitios web, alcanzando de una forma integral y global a todos los elementos y páginas que lo componen. Esta familia la componen diversas herramientas, desde los más conocidos analizadores de páginas a los sistemas de monitorización o una herramienta para la realización de observatorios. El nexo común a todos son los motores de análisis, que han ido creciendo en funcionalidades desde el de accesibilidad a los de calidad web, SEO o movilidad.

El analizador de páginas es:

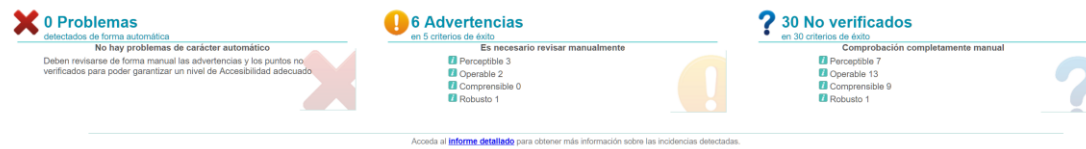
Permite seleccionar el nivel de Accesibilidad a evaluar y las tecnologías empleadas. Como resultado del análisis nos proporciona un informe resumido y otro completo sobre la accesibilidad de los sitios web.



Para ver el detalle de los errores y advertencias se acude al informe detallado, en este caso nos aparece un error pero se trata de un “falso positivo” <http://iktforalle.no/accessibility-checker-evaluation/tawdis.html>

Perceptible	Operable	Comprehensible	Robusto
La información y el manejo de la interfaz de usuario debe ser comprensible.			
Tipología			
Comprobación		Técnicas	Resultado
3.1.1 - Idioma de la página			
Página web	Idioma declarado e idioma real	[H52]	✗
3.1.2 - Idioma de las partes			
Página web	Cambios en el idioma	[H58]	?
3.1.3 - Palabras inusuales			
Página web	Palabras con significados inusuales	[G101]	?
3.1.4 - Abreviaturas			
Estructura y semántica	Abreviaturas y acrónimos	[G97 G55 G62 H60 H28 G70]	?
3.1.5 - Nivel de lectura			
Página web	Nivel de lectura	[G86 G103 G79 G153 G160]	?
3.1.6 - Pronunciación			
Página web	Pronunciación	[G120 G121 G62 H62 G163]	?
3.2.1 - Al recibir el foco			
Scripts	Cambio de contenidos con el evento 'onfocus'	[G107]	?
	Cambios inesperados del foco en el evento 'onfocus'	[F55]	?

En general el resultado tiene que ser 0 problemas en el análisis automático.

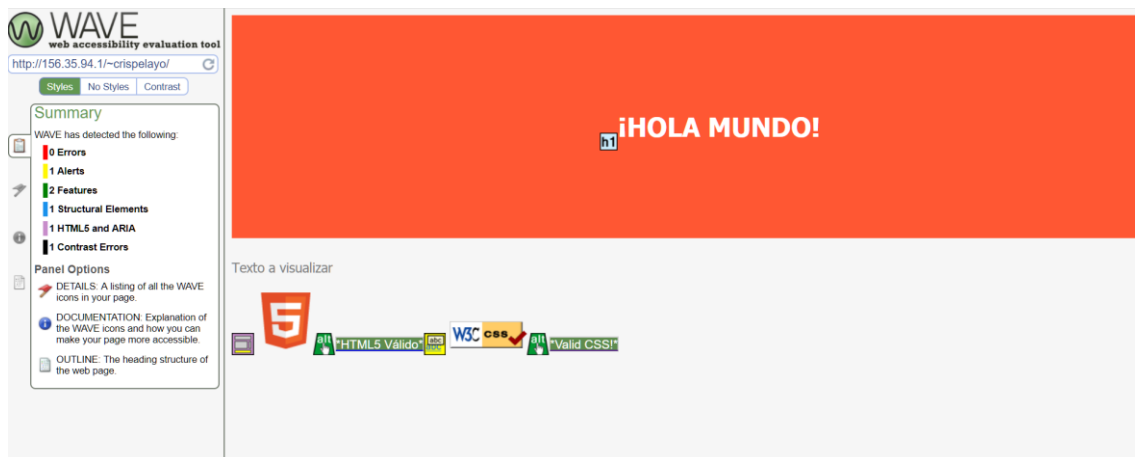


Herramienta Wave - <http://wave.webaim.org/>

WAVE es una herramienta que ayuda a los desarrolladores web a hacer su contenido web más accesible, porque ayuda a evaluar la accesibilidad del contenido web.



Proporciona un informe “visual” de los problemas de accesibilidad detectados.



En general debemos garantizar que el número de errores sea 0.

Herramienta aChecker - <https://achecker.ca>

Esta herramienta evalúa la conformidad de páginas HTML con los estándares de accesibilidad y asegura que el contenido pueda ser accesible por cualquiera.

[Login](#) [Register](#)

Web Accessibility Checker

Web Accessibility Checker


Check Accessibility By:

Web Page URL **HTML File Upload** **Paste HTML Markup**

Address:

Check It

[Options](#)



SEO website tester


¡Localiza todos los problemas en tu página web!

semrush.com

Welcome to **AChecker**. This tool checks single HTML pages for conformance with accessibility standards to ensure the content can be accessed by everyone. See the Handbook link to the upper right for more about the Web Accessibility Checker.

Translate to [English](#) | [German](#) | [Italiano](#)

Web site engine's code is copyright © 2011


[Web Service API](#)

Proporciona un informe organizado por tipo de problema donde se especifica el elemento y la línea que produce el problema, así como la explicación del mismo.

Accessibility Review

Accessibility Review (Guidelines: **WCAG 2.0 (Level AA)**)

Export Format: **PDF** Report to Export: **All** [Get File](#)

Known Problems (1) **Likely Problems (0)** **Potential Problems (24)** **HTML Validation** **CSS Validation**

1.4 Distinguishable: Make it easier for users to see and hear content including separating foreground from background.

Success Criteria 1.4.3 Contrast (Minimum) (AA)

Check 301: The contrast between the colour of text and its background for the element is not sufficient to meet WCAG2.0 Level AA.

Repair: Use a colour contrast evaluator to determine if text and background colours provide a contrast ratio of 4.5:1 for standard text, or 3:1 for larger text. Change colour codes to produce sufficient contrast. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/visual-audio-contrast-contrast.html#visual-audio-contrast-contrast-resources-head>

Line 16, Column 5:

`<p>Texto a visualizar</p>`

Fixed size example: color contrast example

Real size example (12 points): color contrast example

CSS rules for the element:

External CSS (<http://156.35.94.1/~crispelajo/estilo.css>):

```
body{
    font:400 15px Tahoma, sans-serif;
    line-height:1.8;
    color:#818181;
    background-color:#f6f6f6;
}
```

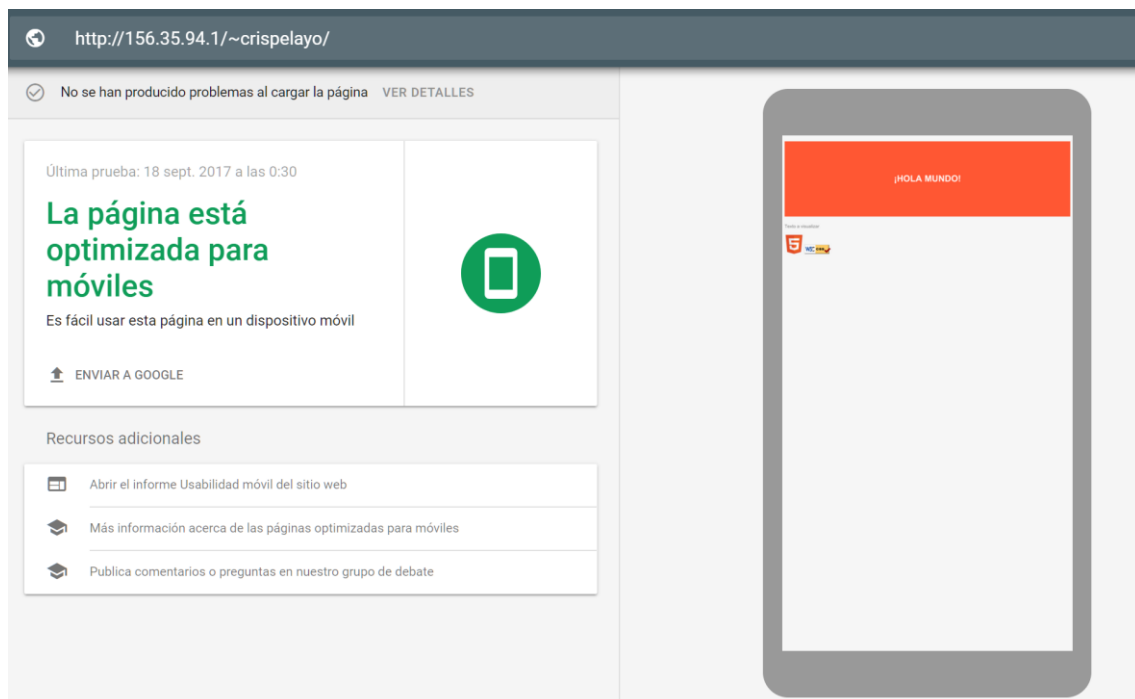
Test de adaptabilidad (*responsive*)

Con la popularización del uso de los dispositivos móviles debemos ser conscientes de cómo es la visualización de los sitios web en diferentes tamaños de pantalla. Para ellos vamos a utilizar dos herramientas que permiten comprobar si el diseño es *responsive* o no.

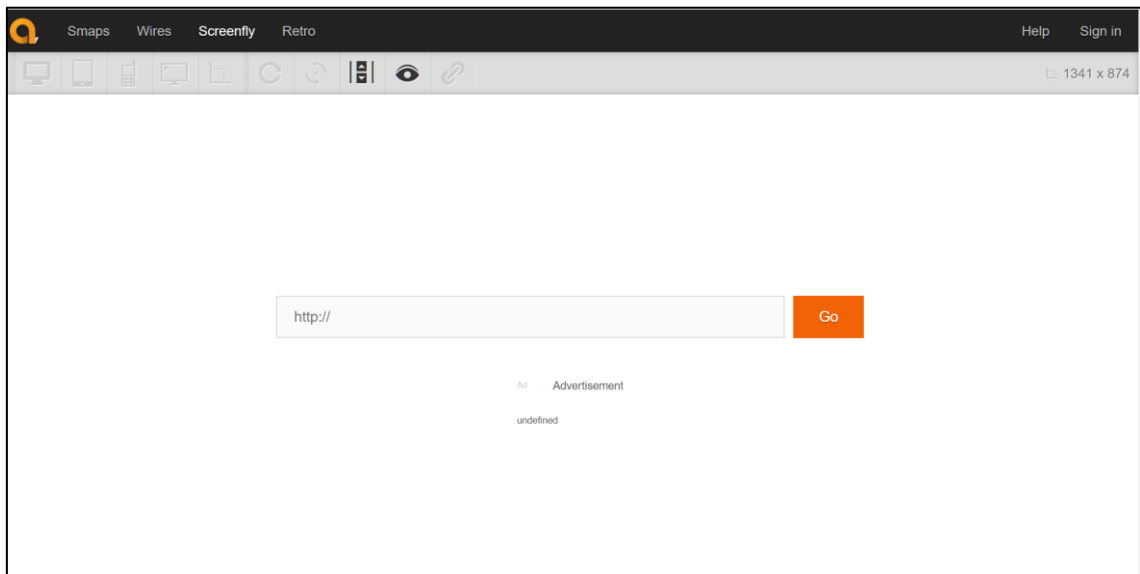
Google Mobile-friendly – <https://search.google.com/test/mobile-friendly>



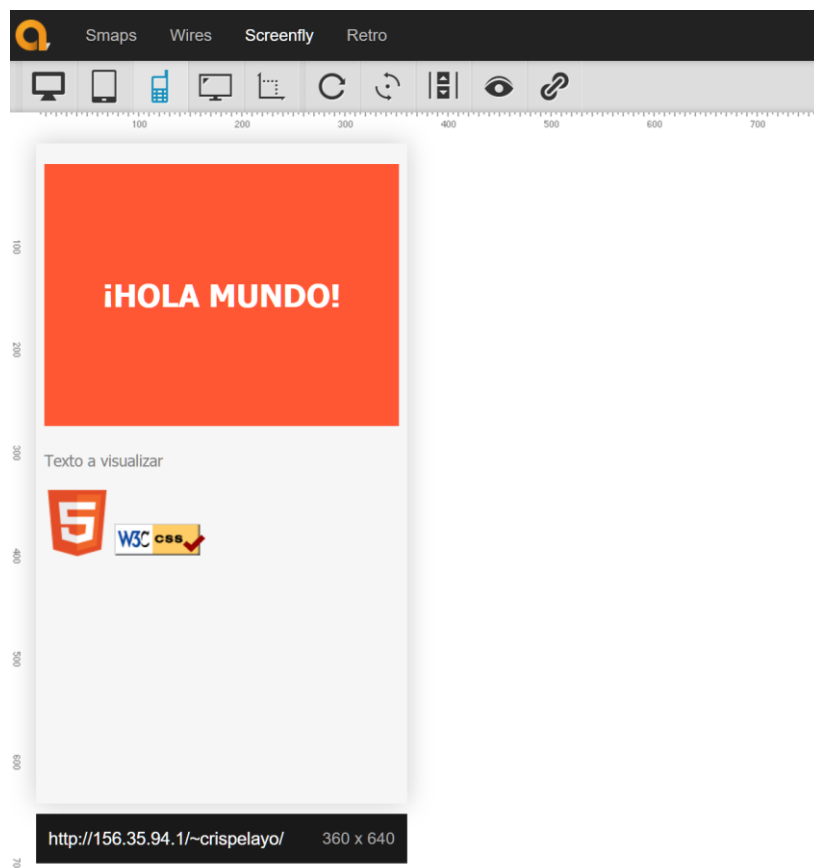
El resultado de la prueba:



Screenfly - <http://quirktools.com/screenfly/>



Nos permite simular la visualización del Sitio web en multitud de soportes de diferentes tipos y marcas.



Anexo I. Recordatorio de permisos en sistemas UNIX

En los sistemas operativos Unix o Linux los directorios y archivos tienen tres tipos de permisos que pueden ser modificados.

Los permisos posibles son:

- r: lectura
- w: escritura
- x: ejecución

Los permisos de un directorio o archivo se otorgan o suprimen sobre tres tipos de usuarios:

- Propietario, en este caso cada alumno (UOxxxx)
- Grupo, en nuestro caso pertenecemos al grupo “students”
- Resto, se refiere al resto de posibles usuarios del servidor

Comprobamos los permisos del directorio personal, por ejemplo:

```
$ls -al
```

```
-rwxr-xr-- 1 propietario grupo 266 sep 12 20:05 miFichero  
drwxr-xr-x 2 propietario grupo 4096 sep 11 13:12 public_html
```

Los elementos más importantes de cada fila son:

- -rw-r--r-- Permisos
- propietario Nombre del propietario del archivo o directorio (UOxxxxx)
- grupo Nombre del grupo al que pertenece el propietario (students)
- nombreFichero Nombre del fichero

La primera columna contiene 10 caracteres que representan los permisos e información sobre el tipo de archivo. Mas explícitamente:

- Primera posición: representa el tipo de archivo (“-” es un archivo, “d” es un directorio)
- Tres posiciones siguientes (2-4): permisos otorgados al propietario
- Tres posiciones siguientes (5-7): permisos otorgados al resto de los componentes del grupo (students)
- Tres posiciones siguientes (8-10): permisos otorgados al resto de usuarios

Por ejemplo, el archivo “miFichero” otorga permisos de lectura, escritura y ejecución al propietario (rwx), lectura y ejecución (r-x) a los miembros del grupo del propietario y de lectura (r-) al resto de los usuarios.

Para otorgar los permisos utilizamos instrucción

```
$chmod XYZ nombreArchivo
```

Donde:

- X: representa los permisos otorgados al propietario
- Y: representa los permisos otorgados al grupo
- Z: representa los permisos otorgados al resto de usuarios

Para dar los permisos se utiliza la notación octal de forma que:

- permiso de lectura (r) valor 4
- permiso de escritura (w) valor 2
- permiso de ejecución (x) valor 1

Ejemplo:

```
$chmod 701 public_html
```

Otorga permisos de lectura, escritura y ejecución ($7 = 4 + 2 + 1$) al propietario, ningún permiso al grupo (0) y permiso de ejecución al resto de los usuarios (1).