

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

Control de autenticación y
acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)



E-mail:

Contraseña:

[Deseo registrarme](#)
[He olvidado mis datos de acceso](#)

Estás en: [Inicio](#) » [Tutoriales](#) » [Creando un tema en Liferay 6.0.6](#)



DESARROLLADO POR:
 [Carlos León Villamayor](#)

Consultor tecnológico de desarrollo de proyectos informáticos.
Constructor de Adictos Al Trabajo 2.0

Ingeniero Técnico en Informática (cursando grado superior)

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE



Fecha de publicación del tutorial: 2011-08-05



[Regístrate para votar](#)     

Creando un tema en Liferay 6.0.6

0. Índice de contenidos.

- 1. Entorno
- 2. Introducción
- 3. Crear nuevo proyecto Liferay
- 4. Identificación de componentes
- 5. Plantillas Velocity
- 6. Customizando el JavaScript y CSS
- 7. Empaquetado
- 8. Conclusiones

1. Entorno

Este tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil 2,4 Ghz Intel Core2 Duo P8600
- Sistema Operativo: Windows 7 Profesional x64 SP1
- Instalada la versión de Liferay Portal 6.0.6 ([versión Integrada en Tomcat 6](#))
- Eclipse Indigo (Revisar [tutorial de Alex](#) para su instalación)
- LiferayIDE Instalado y configurado en Eclipse (Revisar [tutorial de Rubén](#))
- Versión de java SDK 6 instalada en el sistema
- Plugin de Eclipse de editor de Velocity [Veloelise](#) (Opcional)

2. Introducción

En este tutorial veremos la estructura de un tema en Liferay 6.

3. Crear nuevo proyecto Liferay

Después de descargar e instalar el Liferay IDE, creamos un nuevo proyecto de Liferay siguiendo los siguientes pasos:

Catálogo de servicios
[Autentia](#)



Últimas Noticias

-  [Pirineos on Tour](#)
-  [VII Autentia Cycling Day](#)
-  [Autentia patrocina la charla sobre Java SE 7 en Madrid](#)
-  [Alfresco Day 2011](#)
-  [XVII Charla Autentia - Grails - Vídeos y Material](#)

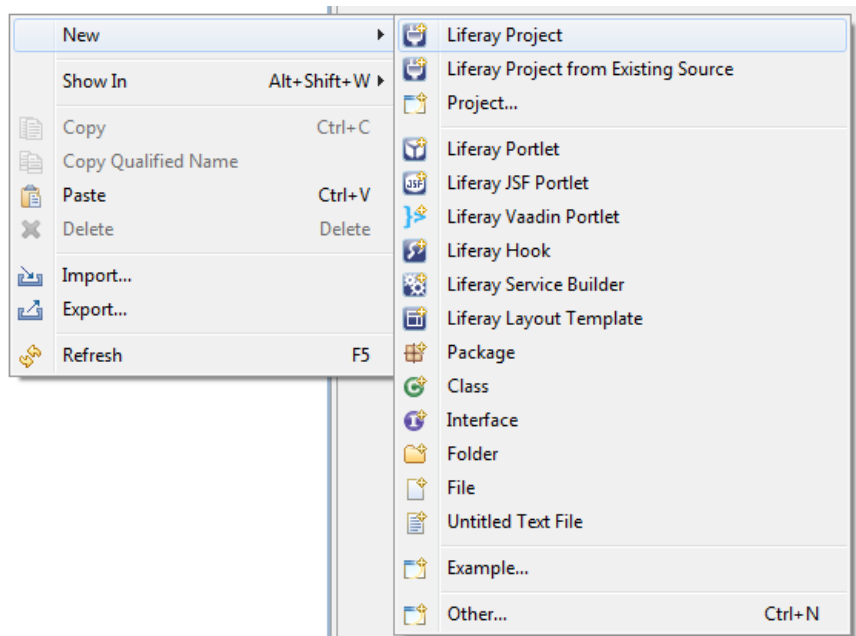
 [Histórico de NOTICIAS](#)

Últimos Tutoriales

-  [Usando CallableStatements para ejecutar procedimientos almacenados](#)
-  [Android: Leer correos de Gmail](#)
-  [Exportación del contenido de un dataTable de JSF con el soporte de Primefaces](#)
-  [Spring schedule annotations](#)
-  [Primeros pasos con Google+](#)

Últimos Tutoriales del Autor

-  [Primeros pasos con Google+](#)



New Liferay Project

Liferay Plugin Project
Create a new plugin project for Liferay Portal.

Project name:

Display name:

[Create a new project from existing sources...](#)

Configuration

Liferay Plugins SDK: [Configure](#)

Liferay Portal Runtime: [New ...](#)

[Advanced project configuration...](#)

Plugin Type

☐ Portlet Create a web application using the portlet framework.

☐ Hook Override or extend Liferay's default behavior and functionality.

☐ Ext Light-weight extension environment for Liferay as a plugin.

☐ Layout Create a new custom layout for Liferay pages.

☒ Theme Build a custom look and feel for the portal.

Working sets

☐ Add project to working sets

Working sets: [Select...](#)

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

Una vez creado añadiremos nuestro servidor:

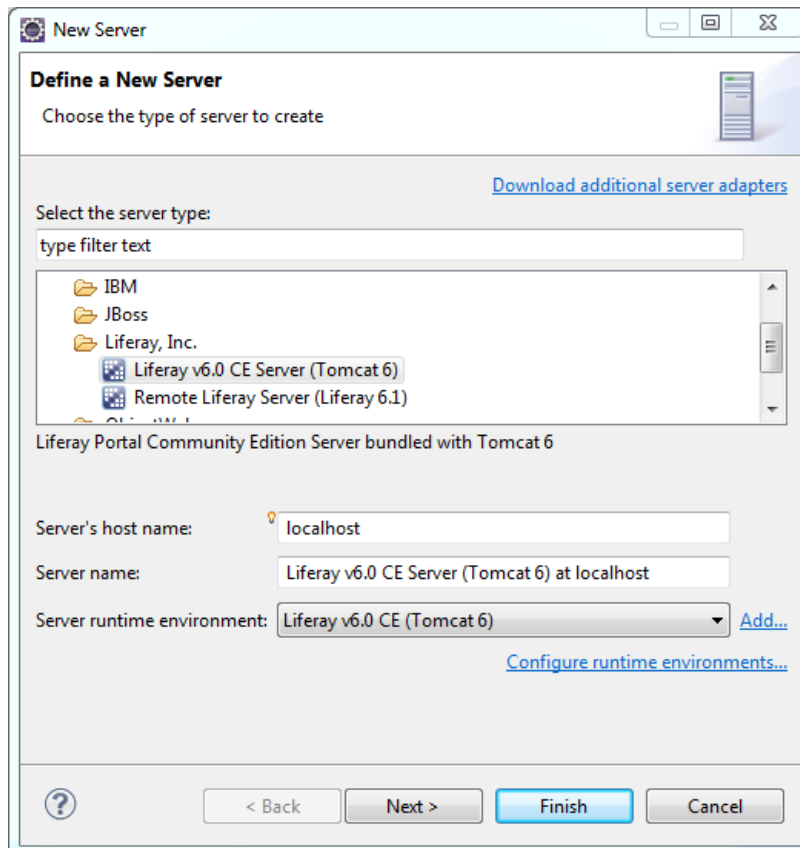
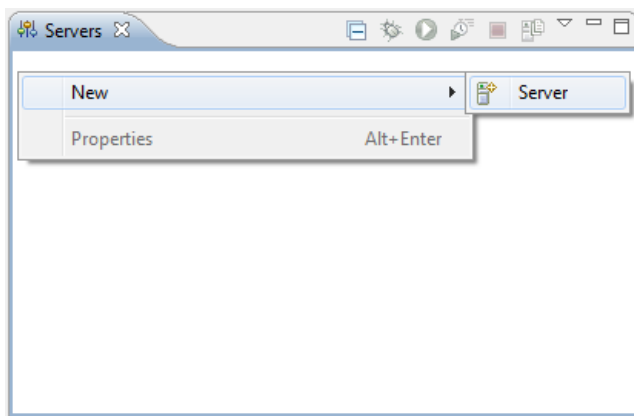
- Google
- Trabajando con JAXB y Eclipse
 - Liferay 6: Cómo crear nuevos tipos de contenido
 - Incluir Gadgets en Liferay 6.0.5: Cómo añadir Gadgets de forma sencilla
 - Instalación de Apache Geronimo en Windows, e Integración con Eclipse Europa

Síguenos a través de:

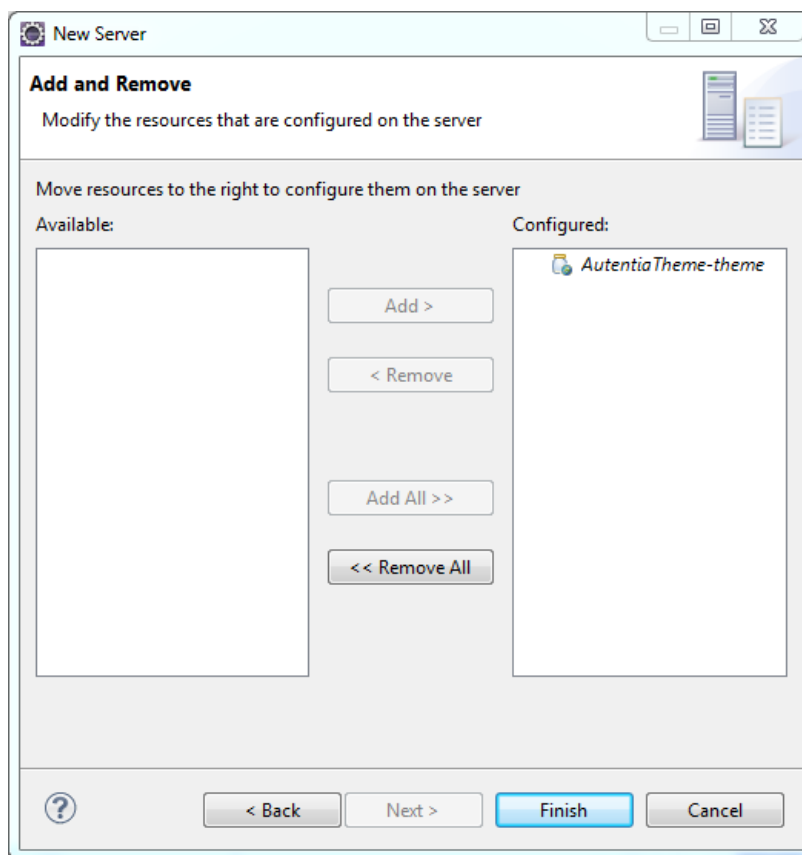


Últimas ofertas de empleo

- 2011-07-06 Otras Sin catalogar - LUGO.
- 2011-06-20 Comercial - Ventas - SEVILLA.
- 2011-05-24 Contabilidad - Especialista Contable - BARCELONA.
- 2011-05-14 Comercial - Ventas - TARRAGONA.
- 2011-04-13 Comercial - Ventas - VALENCIA.



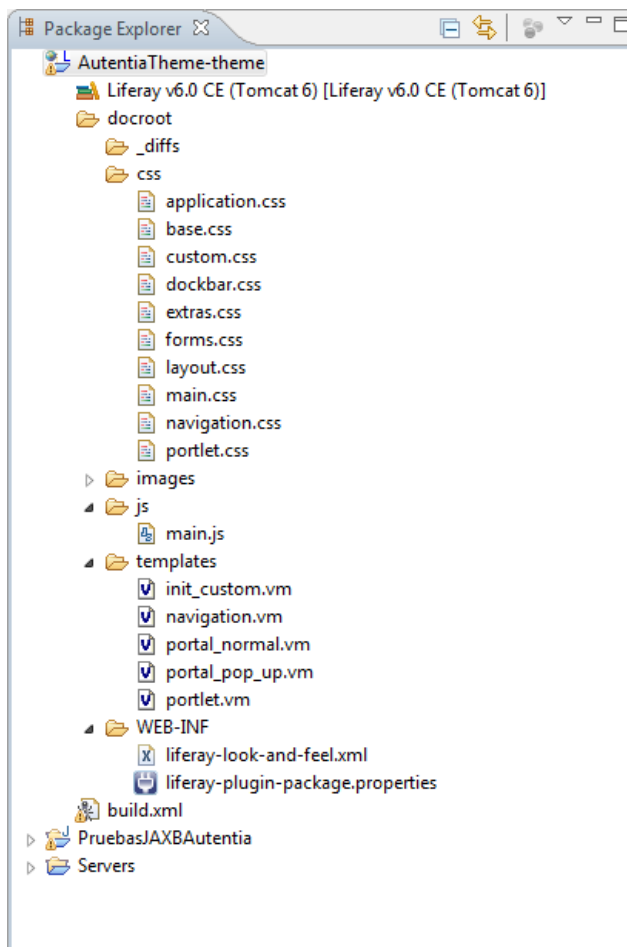
Y añadimos nuestro tema creado:



Ahora ya estamos preparados para trabajar con el tema.

4. Identificación de componentes

Revisando el proyecto:



Identificamos las carpetas siguientes:

- **_diffs**: Es el directorio donde trabajaremos normalmente para **añadir** nuestros diseños (css,js) o imagenes
- **css**: Es donde se ubicarán todos los estilos del tema
- **images**: Es donde se ubicarán todas las imagenes del tema
- **js**: Es donde se ubicarán los script javascript del tema
- **templates**: Es donde se ubicarán las plantillas velocity del tema

- **WEB-INF/liferay-look-and-feel.xml**: Es donde se define el identificador del tema en Liferay y la versión
- **WEB-INF/liferay-plugin-package.properties**: Es donde se definen las características del paquete

5. Plantillas Velocity

Ahora revisamos para que vale cada plantilla comenzando por **portal_normal.vm**, ya que es la plantilla que define el esqueleto de nuestro portal:

```
<!DOCTYPE html>

#parse ($init)

<html class="#language("lang.dir")" dir="#language("lang.dir")" lang="$w3c_language_id">
<head>
    <title>$the_title - $company_name</title>

    $theme.include($top_head_include)
</head>

<body class="$css_class">

    #if($is_signed_in)
        #dockbar()
    #end

    <div id="wrapper">
        <a href="#main-content" id="skip-to-content">#language("skip-to-content")</a>

        <header id="banner" role="banner">
            <hgroup id="heading">
                <h1 class="company-title">
                    <a class="logo" href="$company_url" title="#language("go-to") $company_name">
                        <span>$company_name</span>
                    </a>
                </h1>

                <h2 class="community-title">
                    <a href="$community_default_url" title="#language("go-to") $community_name">
                        <span>$community_name</span>
                    </a>
                </h2>

                <h3 class="page-title">
                    <span>$the_title</span>
                </h3>
            </hgroup>

            #if(!$is_signed_in)
                <a href="$sign_in_url" id="sign-in" rel="nofollow">$sign_in_text</a>
            #end

            #if ($update_available_url)
                <div class="popup-alert-notice">
                    <a class="update-available" href="$update_available_url">#language("updates-are-available-for-liferay")</a>
                </div>
            #end

            #if ($has_navigation)
                #parse ("{$full_templates_path}/navigation.vm")
            #end
        </header>

        <div id="content">
            <nav class="site-breadcrumbs" id="breadcrumbs">
                <h1>
                    <span>#language("breadcrumbs")</span>
                </h1>

                #breadcrumbs()
            </nav>

            #if ($selectable)
                $theme.include($content_include)
            #else
                $portletDisplay.recycle()

                $portletDisplay.setTitle($the_title)

                $theme.wrapPortlet("portlet.vm", $content_include)
            #end
        </div>

        <footer id="footer" role="contentinfo">
            <p class="powered-by">
                #language("powered-by") <a href="http://www.liferay.com" rel="external">Liferay</a>
            </p>
        </footer>
    </div>

</body>

$theme.include($bottom_include)

</html>
```

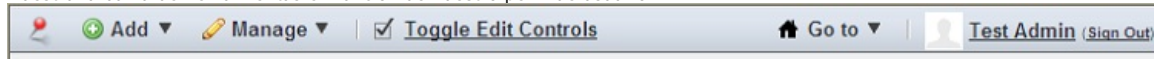
- **Parseando el fichero de inicialización de la plantilla**: El código `#parse ($init)` procesa la inicialización del archivo `{PORTAL_ROOT_HOME}/html/themes/_unstyled/templates/init.vm`. Esta acción inicializa las variables y propiedades necesitadas por los temas y los portlets que usen este tema.
- **CSS y JavaScript includes**: El código `$theme.include($top_head_include)` es el encargado de usar el css y js adecuados para el dispositivo cliente: `{PORTAL_ROOT_HOME}/html/common/themes/top_head.jsp` para navegadores normales o `{PORTAL_ROOT_HOME}/wap/common/themes/top_head.jsp` para navegadores móviles. La sentencia `$theme.include($bottom_include)` incluye el pie de página llamando a los componentes `{PORTAL_ROOT_HOME}/html/common/themes/bottom.jsp` o

`{PORTAL_ROOT_HOME}/wap/common/themes/bottom.jsp`. Entre los tags `</body>` y `</html>` se suelen ejecutar los script js que puedan realizar operaciones despues de la carga de la página, como pueden ser Google analytics o similares.

- **Barra de herramientas:** El siguiente código:

```
1 #if($is_signed_in)
2   #dockbar()
3 #end
```

Muestra la barra de herramientas en función de nuestro perfil de usuario:



- **Cabecera:** El siguiente código:

```
01 <header id="banner" role="banner">
02   <hgroup id="heading">
03     <h1 class="company-title">
04       <a class="logo" href="$company_url" title="#language("go-to") $company_name">
05         <span>$company_name</span>
06       </a>
07     </h1>
08   </hgroup>
09   <h2 class="community-title">
10     <a href="$community_default_url" title="#language("go-to") $community_name">
11       <span>$community_name</span>
12     </a>
13   </h2>
14   <h3 class="page-title">
15     <span>$the_title</span>
16   </h3>
17 </header>
18 ...
19 </header>
```

Muestra la parte relacionada con la compañía o la comunidad del portal en el que estamos navegando.

- **Menús:** El siguiente código:

```
1 <header id="banner" role="banner">
2   ...
3   #if ($has_navigation)
4     #parse ("{$full_templates_path/navigation.vm")
5   #end
6 </header>
```

Muestra el menú contextual:

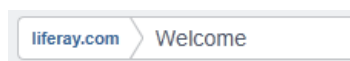


basado en la plantilla:

```
<nav class="sort-pages modify-pages" id="navigation">
  <h1>
    <span>#language("navigation")</span>
  </h1>
  <ul>
    #foreach ($nav_item in $nav_items)
      #if ($nav_item.isSelected())
        <li class="selected">
          <a href="$nav_item.getURL()" $nav_item.getTarget()>#language("navigation")</a>
        </li>
      #else
        <li>
          <a href="$nav_item.getURL()" $nav_item.getTarget()>#language("navigation")</a>
        </li>
      #end
    #foreach ($nav_child in $nav_item.getChildren())
      #if ($nav_child.isSelected())
        <li class="selected">
          <a href="$nav_child.getURL()" $nav_child.getTarget()>#language("navigation")</a>
        </li>
      #else
        <li>
          <a href="$nav_child.getURL()" $nav_child.getTarget()>#language("navigation")</a>
        </li>
      #end
    #end
  </ul>
</nav>
```

- **Rastro de Migas:** El código siguiente muestra el resto de migas:

```
1 <nav class="site-breadcrumbs" id="breadcrumbs">
2   <h1>
3     <span>#language("breadcrumbs")</span>
4   </h1>
5   #breadcrumbs()
6 </nav>
```



- **Portlet:** el siguiente código muestra el contenido de los portlet, y su marco:

```
1 #if ($selectable)
2   $theme.include($content_include)
3 #else
4   $portletDisplay.recycle()
5   $portletDisplay.setTitle($the_title)
6   $theme.wrapPortlet("portlet.vm", $content_include)
7 #end
```

- **Pie de página:** El siguiente código muestra el pie de página

```

1 <footer id="footer" role="contentinfo">
2   <p class="powered-by">
3     #language("powered-by") <a href="http://www.autentia.com" rel="external">Autentia</a>
4   </p>
5 </footer>

```

La plantilla **portal_pop_up.vm** muestra los popups de liferay:

```

<!DOCTYPE html>

#parse ($init)

<html dir="#language ("lang.dir")" lang="$w3c_language_id">

<head>
  <title>$the_title</title>

  $theme.include($top_head_include)

  #css ($css_main_file)
  #js ($js_main_file)
</head>

<body class="portal-popup $css_class">

$theme.include($content_include)

$theme.include($bottom_ext_include)

</body>
</html>

```

La plantilla **portlet.vm** muestra el contenido de un portlet y su marco, para cada una de las vistas del portlet (VIEW, EDIT, HELP):

```

#set ($portlet_display = $portletDisplay)

#set ($portlet_id = $htmlUtil.escapeAttribute($portlet_display.getId()))
#set ($portlet_title = $portlet_display.getTitle())
#set ($portlet_back_url = $htmlUtil.escape($portlet_display.getURLBack()))

<section class="portlet" id="portlet_$portlet_id">
  <header class="portlet-topper">
    <h1 class="portlet-title">
      $theme.iconPortlet() <span class="portlet-title-text">$portlet_title</span>
    </h1>

    <menu class="portlet-topper-toolbar" id="portlet-topper-toolbar_$portlet_id" type="toolbar">
      #if ($portlet_display.isShowBackIcon())
        <a href="$portlet_back_url" class="portlet-icon-back">#language("return-to-full-page")</a>
      #else
        $theme.iconOptions()
        $theme.iconMinimize()
        $theme.iconMaximize()
        $theme.iconClose()
      #end
    </menu>
  </header>

  <div class="portlet-content">
    $portlet_display.writeContent($writer)
  </div>
</section>

```

6. Customizando el JavaScript y CSS

Es recomendable no sobrescribir los archivos que vienen por defecto, sino usar el directorio `/docroot/_diffs` y replicar los subdirectorios `"css"` y `"js"` añadiendo nuestros estilos y/o script.

7. Empaquetado

El IDE de eclipse nos realiza la tarea de deploy directamente en el servidor Tomcat, pero si queremos llevarlo a otro servidor, tan solo deberemos comprimir el contenido del directorio "docroot" en un archivo WAR y podríamos desplegarlo en otro servidor.

8. Conclusiones

Ahora que conocemos la estructura de los temas, tan solo tenemos que adaptar las plantillas, estilos y javascript que hemos visto a nuestras necesidades/requerimientos de una forma sencilla.

Cualquier duda o sugerencia podeis comentarlo.

Saludos.

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Enviar comentario

(Sólo para usuarios registrados)

» **Registrate** y accede a esta y otras ventajas «

COMENTARIOS



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Copyright 2003-2011 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

W3C XHTML 1.0

W3C CSS

XML RSS

XML ATOM