

¿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)



Empieza el jaleo: publicamos el libro

INFORMÁTICA PROFESIONAL

LAS REGLAS NO ESCRITAS PARA TRIUNFAR EN LA EMPRESA

-¿Crees que la informática es sólo programar?
 -¿Sabrías organizar eficientemente un equipo?
 -¿Quién te resuelve las dudas sobre esta profesión?..... o te siembra más?

Hosting patrocinado por **ENREDADOS**

Inicio **Quiénes somos** **Tutoriales** **Formación** **Comparador de salarios** **Comentar libro** **Charlas** **Más**

Estas en: **Inicio** **Tutoriales** Introducción a Tapestry 5**Últimas Noticias**

- » Publicado el primer libro de TDD en castellano de Carlos Blé.
- » Impresiones y video de asistentes al primer coding-dojó de agilismo.es
- » Comentando: Esta no es mi empresa: El desapego de los profesionales del siglo XXI de Ignacio Muro Benayas.
- » Tarta y 7º aniversario de AdictosAlTrabajo.com
- » Formación por calendario
- » Comentando el libro: Escenarios y futuros posibles de...

+ Noticias Destacadas

- » Impresiones y video de asistentes al primer coding-dojó de agilismo.es
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Autentia patrocina el primer Agile Open Spain
- » Lanzamiento del nuevo Web de Autentia

+ Comentarios Cómic**+ Enlaces**
Catálogo de servicios Autentia (PDF 6,2MB)


En formato comic...

Tutorial desarrollado por**Rubén Díaz-Heredero Aguilera**

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero en Informática, especialidad en Ingeniería del Software

Puedes encontrarme en [Autentia](#)

Somos expertos en Java/J2EE

Catálogo de servicios de Autentia

Descargar (6,2 MB)

Descargar en versión comic (17 MB)

[AdictosAlTrabajo.com](#) es el Web de difusión de conocimiento de [Autentia](#).



[Catálogo de cursos](#)

Descargar este documento en formato PDF: [IntroduccionTapestry.pdf](#)

Acceso de usuarios registrados:E-mail: Contraseña:

[Deseo registrarme](#)

[He olvidado mis datos de acceso](#)

Registra tu empresa:

Descubre las ventajas de registrar tu empresa en [AdictosAlTrabajo...](#)

[Registrar mi empresa](#)

[Listado de empresas ya registradas](#)

Fecha de creación del tutorial: 2010-01-17

Introducción a Tapestry 5.**0. Índice de contenidos.**

1. ¿Qué es Tapestry?
2. Entorno.
3. Crear primer proyecto Tapestry.
4. Creación de la primera página.
5. Creación del primer componente.
6. Conclusiones.

1. ¿Qué es Tapestry?

Es un framework web de código abierto pensado para programadores Java que trata de dar solución a los dos problemas fundamentales del desarrollo web con Java: los JSP, que no respetan el modelo vista controlador, al permitirnos introducir cualquier código Java dentro de la vista de las páginas y la cantidad de configuración en XML o con anotaciones que se necesitan en otros frameworks para comenzar a desarrollar con ellos.

Esta solución se la tenemos que agradecer a la persona de Howard Lewis Ship quien, basándose en la idea del code behind de .NET, donde cada página lleva un archivo asociado con los elementos que la componen y la lógica asociada a esos elementos que se disparan en forma de eventos, construyó este framework para poder utilizar esta idea en Java.

Además es un framework que no presenta una fuerte curva de aprendizaje y que al estar orientado a componentes lo hace altamente productivo y reutilizable. Presenta varios componentes de serie que nos facilitan mucho las tareas típicas de cualquier aplicación web como la visualización de resultados paginados, la validación de formularios o la internacionalización entre otras y se integra a la perfección con otros frameworks muy conocidos como Spring e Hibernate.

2. Entorno

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Asus G50Vseries (Core Duo P8600 2.4GHz, 4GB RAM, 320 GB HD).
- Sistema operativo: Ubuntu 9.04 Desktop
- Mozilla Firefox 2.0.0.20
- Maven 2.2.1
- Tapestry 5.1.0.5

3. Crear primer proyecto Tapestry

Para crear nuestro primer proyecto en Tapestry vamos a utilizar Maven 2, como no podría ser de otra forma, con lo que sólo tendremos que poner en un terminal lo siguiente:

```
01. | mvn archetype:generate -DarchetypeCatalog=http://tapestry.formos.com/maven-repository
```

[Anuncios Google](#) [Mail Exchange Server](#) [Fichero De Clientes](#) [PC Fax Software](#) [Filter](#) [Ficheros DWG](#)



Web

[www.adictosaltrabajo.com](#)

Últimos tutoriales

2010-01-17
[Introducción a Tapestry 5](#)

2010-01-14
[JMeter. Gestión de usuarios](#)

2010-01-14
[Patrón Visitor con commons-collections y sus Closures](#)

2010-01-12
[Creación de servicios web Restful, con soporte a persistencia, en NetBeans.](#)

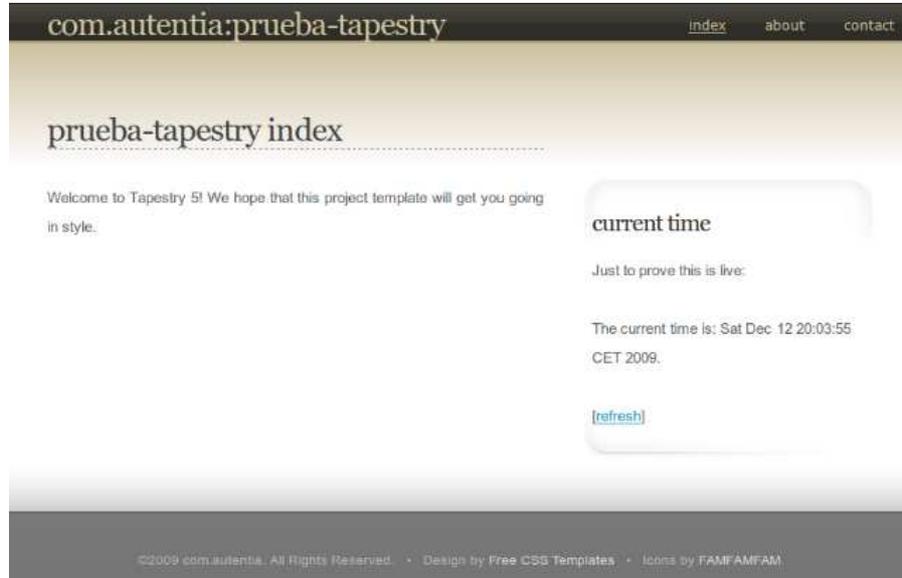
2010-01-11

De esta forma en la consola se nos va a pedir que introduzcamos el número del arquetipo que vamos a utilizar, en nuestro caso el único que hay el 1. Y luego añadimos la información que necesita Maven. Por ejemplo, podemos establecer estos datos:

- **groupId:** com.autentia
- **artifactId:** prueba-tapestry
- **version:** por defecto
- **package:** por defecto

Una vez creado el proyecto lo más cómodo es importarlo a algún IDE que nos facilite la edición del código. Nosotros lo vamos a importar a Eclipse con la ayuda del plugin para Maven.

Esto nos proporciona una aplicación web que directamente podemos ejecutar en un servidor Apache Tomcat desde el IDE que queramos utilizar. La apariencia de la aplicación sin tocar una sola línea de código es esta:



Veamos como esta estructurado el proyecto recién creado:

[JMeter y JSF. Extracción del parámetro ViewState](#)

2010-01-07
[Importar el correo de Microsoft Outlook al cliente de correo de Mac OS.](#)

2010-01-07
[Monitor de Hudson para Eclipse.](#)

2010-01-07
[Patrones de diseño de XML Schema](#)

2010-01-04
[Procesador Inteligente de Eventos \(IEP\) con OpenESB](#)

2010-01-04
[PHP Vs Java](#)

2009-12-29
[Tutorial de BPEL con OpenESB \(II\)](#)

2009-12-29
[Tutorial de BPEL con OpenESB \(I\)](#)

2009-12-28
[Pruebas funcionales de servicios web con soapUI](#)

2009-12-28
[SoapUI: jugando con web services](#)

2009-12-17
[¿Qué son el cloud computing y google app engine?](#)

2009-12-14
[JavaBean Datasource Ireport](#)

2009-12-11
[Contract-First web services con Visual Studio 2008](#)

2009-12-09
[Integrando Sonar con Hudson](#)

2009-12-09
[Apache + Tomcat: Balanceo de carga y alta disponibilidad](#)

2009-12-08
[MySQL: Replicación de bases de datos en MySQL](#)

2009-12-07
[Analizando la calidad del código Java con Sonar](#)

2009-12-03
[Instalar OpenESB 2.1 e Introducción](#)



Lo primero que llama la atención es que no contamos con ficheros con extensión .html sino que trabajamos con ficheros con extensión .tml. Estos ficheros son lo que van a tener el contenido XHTML de los páginas pero con la sintaxis especial de Tapestry.

Luego si nos fijamos un poco más nos damos cuenta de que existen tantas clases Java como ficheros .tml. Esto es así porque cada una de estas clases Java define la parte dinámica de los ficheros. Para aclarar esto vamos a crear nuestra página con Tapestry para que a medida que vayamos publicando más tutoriales se entienda mejor el código que viene por defecto.

En este tutorial sólo comentar que la única configuración XML que encontramos para hacer funcionar el proyecto se encuentra dentro del fichero web.xml y son estas pocas líneas:

```

view plain print ?
01. <?xml version="1.0" encoding="UTF-8"?>
02. <!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN" "http://
03. <web-app>
04.   <display-name>prueba-tapestry Tapestry 5 Application</display-name>
05.   <context-param>
06.     <param-name>tapestry.app-package</param-name>
07.     <param-value>com.autentia</param-value>
08.   </context-param>
09.   <filter>
10.     <filter-name>app</filter-name>
11.     <filter-class>org.apache.tapestry5.TapestryFilter</filter-class>
12.   </filter>
13.   <filter-mapping>
14.     <filter-name>app</filter-name>
15.     <url-pattern>*/</url-pattern>
16.   </filter-mapping>
17. </web-app>

```

De las cuales las más importantes son las referidas al parámetro "tapestry.app-package" donde le indicamos a partir de que nivel de la estructura de paquetes tiene que empezar a buscar las páginas, componentes, etc...

4. Creación de la primera página.

Como no podía ser de otra forma para ilustrar la creación de una página con Tapestry vamos a utilizar un HolaMundo (si tuviera una máquina del tiempo volvería a atrás y patentaría el HolaMundo). Como ya se ha dicho las páginas se componen de una parte visual en XHTML y de un fichero Java asociado.

Primero vamos a crear la página con extensión .tml dentro de la ruta resources/com/autentia/pages y la vamos a llamar HolaMundo.tml con el siguiente contenido:

2009-11-25
[Tutorial de Google Forms](#)

2009-11-23
[Alfresco - Modificando y eliminando contenido desde nuestras aplicaciones Java](#)

2009-11-23
[Alfresco - Añadiendo contenido desde nuestras aplicaciones Java](#)

2009-11-23
[Haciendo funcionar Google Chromium Operating System sobre Windows XP](#)

2009-11-23
[Redimensionar el tamaño de una partición de VirtualBox](#)

2009-11-20
[El Arte de las Presentaciones. Siguiendo la Senda Zen.](#)

2009-11-18
[Tutorial basico de google wave bots](#)

2009-11-13
[Introducción a Escritorios Animado \(Winamp y MilkWave \)](#)

2009-11-12
[Maven Assembly Plugin: empaquetando aplicaciones con Maven para la ejecución de procesos batch.](#)

2009-11-12
[Tutorial básico sobre Google Application Engine](#)

2009-11-11
[Instalación de Glassfish 2.1](#)

2009-11-26
[Rational Software Architect y Rational RequisitePro](#)

2009-10-27
[Cómo conseguir que Subversion avise a Hudson para lanzar una build](#)

2009-10-26
[Cómo instalar Hudson en Apache Tomcat](#)

2009-10-26
[Estilos externos en iReport](#)

2009-10-14
[Echo en MySQL](#)

2009-09-28
[iReport : Solución al problema de los parámetros dinámicos.](#)

```

view plain print ?
01. <html xmlns:t="http://tapestry.apache.org/schema/tapestry_5_1_0.xsd">
02.   <head>
03.     <title>Primera página con Tapestry 5</title>
04.   </head>
05.   <body>
06.     <p>${helloWorld} </p>
07.   </body>
08. </html>
    
```

Ahora vamos a crear la clase Java asociada a esta clase y que va a contener la lógica de recuperar el mensaje de bienvenida con el siguiente código:

```

view plain print ?
01. package com.autentia.pages;
02. public class HolaMundo {
03.     public String getHolaMundo(){
04.         return "Hola Mundo Tapestry 5!!!";
05.     }
06. }
    
```

Ahora si ejecutamos la página de esta forma <http://localhost:8080/prueba-tapestry/holaMundo> veremos que Tapestry responde con el siguiente error, mucho más descriptivo que el típico de Apache Tomcat:

An unexpected application exception has occurred.

java.lang.RuntimeException

Exception assembling root component of page HolaMundo: Could not convert 'helloWorld' into a component parameter binding: Class com.autentia.pages.HolaMundo does not contain a property named 'helloWorld' (within property expression 'helloWorld'). Available properties: class, componentResources, holaMundo.

org.apache.tapestry5.ioc.internal.util.TapestryException

Could not convert 'helloWorld' into a component parameter binding: Class com.autentia.pages.HolaMundo does not contain a property named 'helloWorld' (within property expression 'helloWorld'). Available properties: class, componentResources, holaMundo.

location:

classpath:com/autentia/pages/HolaMundo.tml, line 8

3	xmlns:p="tapestry:parameter">
4	<head>
5	<title>Tapestry 5 Demo Application</title>
6	</head>
7	<body>
8	<p>\${helloWorld}</p>
9	</body>
10	</html>

org.apache.tapestry5.internal.services.PropertyExpressionException

Class com.autentia.pages.HolaMundo does not contain a property named 'helloWorld' (within property expression 'helloWorld'). Available properties: class, componentResources, holaMundo.

Como vemos se marca la línea donde esta el error y se ha producido porque la clase Java asociada no contiene una propiedad con el nombre helloWorld y ofrece otras propiedades que si contiene como holaMundo. Entonces editamos el fichero HolaMundo.tml y sustituimos "helloWorld" por "holaMundo" sin reiniciar el servidor, ya que estas es otra de las ventajas que tiene trabajar con Tapestry, no hace falta reiniciar el servidor cuando se modifica un .tml y cuando se modifica la clase Java asociada el servidor se reinicia automáticamente de una forma muy rápida.

La sintaxis \${holaMundo} sirve para indicar a Tapestry que sustituya ese valor buscando en la clase Java asociada un método llamado getHolaMundo() que ahora si que va a encontrar, mostrando el mensaje de bienvenida por pantalla sólo refrescando la página.

5. Creación del primer componente

Un componente en Tapestry es un elemento que se va a poder reutilizar en todo el proyecto. Para ilustrar la creación de nuestro primer componente, vamos a crear un componente de tipo Layout que aplicaremos a la página creada anteriormente. Tenemos un ejemplo en el código que se ha creado por defecto, Layout.tml y Layout.java.

Basándonos en ese ejemplo vamos a crear nuestro propio layout mucho más sencillo y didáctico. Para ello creamos el fichero `src/main/resources/com/autentia/components/LayoutAutentia.tml` con todo el contenido XHTML que queramos, en el caso de este ejemplo este va a ser el contenido:

- 2009-09-27 [Mavenizar Liferay SDK](#)
- 2009-09-27 [Crear un plugin con Liferay SDK](#)
- 2009-09-23 [EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Apache Open EJB.](#)
- 2009-09-21 [Primeros pasos con JBoss Seam](#)
- 2009-09-21 [Integración con sistemas de Bug Tracking desde NetBeans 6.7](#)

rebas de con Maven, bedded JBoss

e Liferay en ente

illa Maven

illa con esPlugin

las de ... ¿de dónde racteres

deploy del site SourceForge

iertas de

2009-07-31 [T. Información - Operador \(dia / noche\) - BARCELONA.](#)

2009-06-25 [Atención a cliente - Call Center - BARCELONA.](#)

2009-06-19 [Otras - Ingeniería \(minas, puentes y puertos\) - VALENCIA.](#)

2009-06-17 [Comercial - Ventas - ALICANTE.](#)

2009-06-03 [Comercial - Ventas - VIZCAYA.](#)

Anuncios Google

```

view plain print ?
01. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD:/
02. <html xmlns="http://www.w3.org/1999/xhtml" xmlns:t="http://tapestry.apache.org/schema/tap
03.     xmlns:p="tapestry:parameter">
04.     <head>
05.         <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
06.         <title>${titulo}</title>
07.     </head>
08.     <body>
09.         <div id="header">
10.             <div id="logo">
11.                 
12.             </div>
13.             <div>
14.                 <p>${autor}</p>
15.             </div>
16.         </div>
17.         <div id="page">
18.             <t:body/>
19.         </div>
20.     </body>
21. </html>

```

Este layout va a mostrar el contenido de la página donde se aplique más un logo de la empresa con el nombre del autor y el título de la página que tienen que ser proporcionados por cada una de las páginas que vayan a utilizar este layout. La etiqueta `<t:body/>` indica donde se va a mostrar el contenido de la página que utiliza el layout.

Ahora vamos a crear el archivo asociado Java en la ruta `src/main/java/com/autentia/components/LayoutAutentia.java` con el siguiente contenido:

```

view plain print ?
01. package com.autentia.components;
02.
03. import org.apache.tapestry5.*;
04. import org.apache.tapestry5.annotations.*;
05. import org.apache.tapestry5.ioc.annotations.*;
06. import org.apache.tapestry5.BindingConstants;
07.
08. public class LayoutAutentia{
09.     @Property
10.     @Parameter(required = true, defaultPrefix = BindingConstants.MESSAGE)
11.     private String titulo;
12.
13.     @Property
14.     @Parameter(required=true, defaultPrefix=BindingConstants.LITERAL)
15.     private String autor;
16. }

```

Dentro de esta clase hemos definido las dos propiedades que utiliza el layout "titulo" y "autor". Las anotaciones que las preceden quieren decir que titulo y autor van a ser una propiedades que obligatoriamente tienen que ser proporcionar las páginas que hagan uso de este componente si no se mostrará un error por pantalla. La diferencia entre ellas es que el valor que se le de a titulo se va a utilizar como key para buscar el valor en el fichero de idioma correspondiente, mientras que, el valor de autor se va a interpretar tal cual.

Para utilizar el componente editamos el fichero `HolaMundo.tml` modificando la cabecera de esta forma:

```

view plain print ?
01. <html t:type="layoutAutentia" title="hola-mundo" autor="Ruben Aguilera" xmlns:t="http://ta

```

Para que se muestre correctamente el título de la página tenemos que tener creado el fichero de idioma correspondiente. Si sólo queremos que este texto este disponible para el ámbito de esta página, crearemos el fichero `src/main/resources/com/autentia/pages/HolaMundo.properties`, si por el contrario queremos que este texto sea accesible desde cualquier parte de la aplicación tendremos que editar el fichero `src/main/webapp/WEB-INF/app.properties`. En cualquiera de los dos casos deberemos introducir la entrada "hola-mundo" con un valor cualquiera, "Esto es un Hola Mundo con Tapestry" y replicado en tantos ficheros como idiomas queramos que soporte la aplicación.

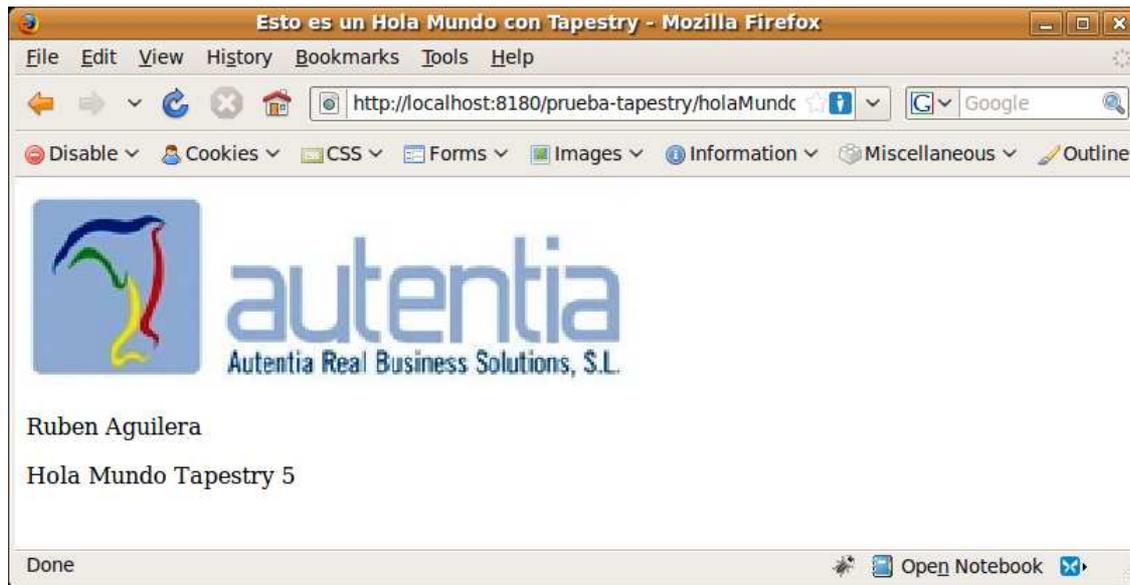
Sólo falta un pequeño detalle para habilitar la internacionalización y es editar el fichero `src/main/java/com/autentia/services/AppModule.java` y buscar el método `contributeApplicationDefaults` para añadir en la configuración todos los idiomas que queramos soportar:

```

view plain print ?
01. public static void contributeApplicationDefaults( MappedConfiguration<String, String> conf:
02.     configuration.add(SymbolConstants.SUPPORTED_LOCALES, "en,es");
03. }

```

En este caso nuestra aplicación va a soportar el inglés y el castellano. Ahora si volvemos a ejecutar la URL `http://localhost:8080/prueba-tapestry/holaMundo` veremos el resultado:



Para aplicar estilos y Javascript a la página tenemos dos formas de hacerlo, la habitual que sería incluyendo las referencias dentro del fichero .tml y al estilo Tapestry con anotaciones en la clase del componente. Por ejemplo en nuestro caso sería:

```

view plain print ?
01. @IncludeStylesheet("context:layout/layout.css")
02. @IncludeJavaScriptLibrary("context:layout/layout.js")
03. public class LayoutAutentia{...}

```

Donde estamos indicando a Tapestry que busque las referencias dentro del contexto colgando de la carpeta layout. Esto es en realidad src/main/webapp/layout.

6. Conclusiones

En este pequeño tutorial sólo hemos podido ver un pequeño esbozo de lo que nos ofrece Tapestry, en posteriores tutoriales vamos a ver como hacer los procedimientos típicos de una aplicación web como: internacionalización, creación de CRUDs con paginación, internacionalización e incluso AJAX de una manera muy sencilla gracias a este estupendo framework. Aconsejo a los más intrépidos e impacientes que hagan pruebas con el proyecto de ejemplo para ir viendo como funciona sobre todo la parte de creación de menús que abordaremos en otro tutorial.

Saludos.

¿Qué te ha parecido el tutorial? Déjanos saber tu opinión y ¡vota!

Muy malo
 Malo
 Regular
 Bueno
 Muy bueno

(Sólo para usuarios registrados)

[» Regístrate y accede a esta y otras ventajas «](#)

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.

(Sólo para usuarios registrados)

[» Regístrate y accede a esta y otras ventajas «](#)

Autor

Mensaje de usuario registrado

- Puedes inscribirte en nuestro servicio de notificaciones haciendo clic [aquí](#).
- Puedes firmar en nuestro libro de visitas haciendo clic [aquí](#).
- Puedes asociarte al grupo AdictosAlTrabajo en XING haciendo clic [aquí](#).
- Añadir a favoritos Technorati. 



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

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com

soluciones para negocic

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
Introducción a Tapestry 5	Si estás cansado de programar con frameworks web con los que para hacer un simple holamundo tienes que tirar muchísimas líneas de código de XML o peor sigues trabajando con JSP donde además metes código Java en forma de snippets que son difícil de depurar	2010-01-17	18	-	-	
Importar el correo de Microsoft Outlook al cliente de correo de Mac OS.	En este tutorial vamos a explicar cómo importar el correo de Windows a Mac, del cliente de correo Microsoft Outlook al cliente de correo Apple Mail de Mac OS X	2010-01-07	262	-	-	
JavaBean Datasource Ireport	La particularidad del caso que nos ocupa, es conseguir que la fuente de datos del informe sea una lista de JavaBeans y no una consulta definida previamente en el informe.	2009-12-14	863	Bueno	1	
Tutorial básico sobre Google Application Engine	En este tutorial veremos cómo ejecutar una simple aplicación web en la infraestructura de Google	2009-11-12	1577	Bueno	1	
Instalación de Glassfish 2.1	En este tutorial nos veremos cómo instalar el servidor de aplicaciones GlassFish. Además veremos los primeros pasos, como entrar en la consola de administración del servidor, y desplegar una aplicación EAR (Enterprise Application)	2009-11-11	2072	Bueno	1	
Joomla 1.5. Instalación y configuración	Veamos en detalle cómo instalar Joomla 1.5 y aplicar algunas configuraciones posteriores de las disponibles en este CMS	2009-08-14	8088	-	-	
Cómo hacer una página web	En este tutorial vamos a aprender a hacer una página web con HTML y publicarla en internet	2009-08-10	10651	-	-	
Funciones esenciales para crear un juego.	En este tutorial, seguimos usando el motor jMonkeyEngine, y hemos intentado enseñar como se crean escenarios y personajes, con sus funciones de rotamiento, y movimiento.	2009-07-30	3523	Bueno	1	
Directorio de ejemplos de jMonkey Engine	Directorio de los ejemplos más relevantes de jMonkeyEngine, que nos ayudarán a crear juegos en 3D	2009-07-20	3324	-	-	
Como implementar el Scene Monitor para analizar las escenas en jMonkeyEngine	Implementación de SceneMonitor de jMonkeyEngine en nuestra aplicación.	2009-07-16	2181	-	-	

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.