

¿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)

 Powered by 	Hosting Patrocinado por enREDados.com 
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------

[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Tutoriales](#) | [Contacte](#)

Lanzamiento TNTConcept

[Autentia](#) da un paso más en su evolución: Lanzamiento de software propio. Ponemos a vuestra disposición el software que hemos construido para nuestra gestión interna, llamado TNTConcept (auTeNTia).

Construida con las últimas tecnologías de desarrollo Java/J2EE (Spring, JSF, Hibernate, Maven, Subversion, etc.) y disponible en licencia GPL, seguro que a muchos profesionales independientes y PYMES os ayudará a organizar mejor vuestra operativa.

Las cosas grandes empiezan siendo algo pequeño Saber más en: <http://tntconcept.sourceforge.net/>

	<p>Tutorial desarrollado por: Javier Antoniucci</p> <p>Puedes encontrarme en Autentia Somos expertos en Java/J2EE Contacta en javier.antoniucci@autentia.com</p>	<p>www.adictosaltrabajo.com es el Web de difusión de conocimiento de www.autentia.com</p>  <p>autentia real business solutions</p> <p>Catálogo de cursos</p>
-----------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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

[Firma en nuestro libro de Visitas](#) <-----> [Asociarme al grupo AdictosAlTrabajo en eConozco](#)

Master Experto Java

100% alumnos trabajando Nuevo temario de Struts + Ajax
www.grupoatrium.com

Descarga software gratis

Saca el máximo rendimiento de tu PC con Google Pack. Infórmate
pack.google.es

Programacion PHP

Desarrollo de Aplicaciones Web
 Programación Web - LAMP - PHP
 MySQL
www.trueologic.com.ar

Master-Java

15 perfiles informáticos remunerado, proyectos estrellas
www.meca97.es

Anuncios Google

URLs amigables con Spring MVC

Las URLs amigables son cada vez más frecuentes y adictivas. Si implementamos en nuestra aplicación este tipo de URLs, a medida que el usuario navega en las operaciones puede visualizar en la barra de navegación URLs tan intuitivas como `miaplicacion.miempresa.com/clientes/misclientes/001/editar` y puede guardar un bookmark para volver a la edición de dicho cliente.

Existe todo un estándar que no sólo abarca las URLs amigables llamado REST (Transferencia de Estado Representacional, ver http://es.wikipedia.org/wiki/Representational_State_Transfer) donde se describe cómo llevar el estado de interacción a la URL, utilización jerárquica de los conceptos, utilización de los verbos PUT/GET/POST del HTTP, etc.

Sin ser tan ambiciosos podemos fácilmente implementar en Spring una solución a esta funcionalidad utilizando los componentes que el framework contempla para estos casos.

Vamos a hacer un ejemplo práctico utilizando Spring MVC y para ello haremos:

- Configuración del URL Handler
- Configuración de los controladores
- Mejora propuesta

Dado que no es el objetivo del tutorial explicar Spring MVC, comenzaremos desde un nuevo proyecto Spring MVC o modificando uno existente.

Configuración del URL Handler

La configuración de la resolución de nombres consiste en incluir en el `applicationContext.xml` (o donde hayamos configurado la aplicación) la definición de un `urlMapping` donde la clase que lo implementa sea `SimpleUrlHandlerMapping` y en mappings los pongamos los patrones de path utilizando asteriscos para las partes que podrían variar cuando se trata de identificadores, alias, etc. de un concepto. Si ningún patrón se ajustará, entonces con `defaultHandler` definimos cual controlador se invocará.

```
<bean id="urlMapping" class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">
    <property name="defaultHandler" ref="clienteListar" />
    <property name="mappings">
```

```

        <props>
            <prop key="/clientes/misclientes/*/editar.html"> clienteEditar</prop>
        </props>
    </property>
</bean>

```

Configuración de los controladores

En el applicationContext.xml (o donde hayamos configurado los controladores) comprobaremos que todos los controladores utilizados en el mappings de urlMapping estén incluidas:

```

<bean id="login" class="com.miempresa.miaplicacion.controller.clienteListar" />
<bean id="login" class="com.miempresa.miaplicacion.controller.clienteEditar" />

```

Y en el propio controlador, en caso de necesitar obtener un valor incluido en una URL (como en /clientes/misclientes/001/editar) siempre podemos utilizar algo como:

```

...
String[] valoresUrl = url.split("/");
String idCliente = valoresUrl[2];
...

```

Mejora propuesta

Como mejora a esta solución, en Autentia hemos implementado una subclase del SimpleUrlHandlerMapping donde implementamos la extracción de los valores de las variables contenidas en los mismos haciéndolos configurables desde una propiedad variablesRestUrls de la clase configurada en el applicationContext.xml.

```

protected Object lookupHandler(final String urlPath,
    final HttpServletRequest req) {
    // Obtener el handler
    FlowController controlador = (FlowController) super.lookupHandler(
        urlPath, req);
    if (controlador == null) {
        controlador = (FlowController) super.getDefaultHandler();
    }
    // Según el flujo seleccionado, extraer los parámetros de la URL e
    // inyectarlos como atributos del request
    final String flujoSeleccionado = controlador.getArgumentHandler().getDefaultFlowId().toLowerCase();
    final String variablesRestUrl = variablesRestUrls.getProperty(flujoSeleccionado);
    if (variablesRestUrl != null) {
        final String[] variables = variablesRestUrl.split("/");
        final String friendlyUrl = urlPath.replace(extensionControlador, "");
        final String[] valoresUrl = friendlyUrl.split("/");
        for (int i = 0; i < variables.length && i < valoresUrl.length; i++) {
            if (!variables[i].equals("") && !variables[i].equals("none")) {
                req.setAttribute(variables[i], valoresUrl[i]);
            }
        }
    }
}

```

```

        }
    }
    return controlador;
}

```

Como vemos, los valores extraídos se introducen como atributo de la request utilizando el nombre indicado.

Un ejemplo de configuración sería:

```

<property name="variablesRestUrls">
    <value>
        login-flow=/none
        pglobal-flow=/none
        cuentas-flow=/none/none
        movimientos-flow=/none/none/aliasCuenta
        transferencias-flow=/none
    </value>
</property>

```

Dado que la programación de un Handler es algo muy sencillo y que customizamos de acuerdo a nuestras necesidades, no hemos incluido copia de nuestra solución particular.

Desde Autentia contamos con los conocimientos y experiencia para ayudarle a aplicar nuevas tecnologías a su negocio.

No dude en contactarse con nosotros mediante www.autentia.com .



This work is licensed under a [Creative Commons Attribution-NonCommercial-No Derivative Works 2.5 License](http://creativecommons.org/licenses/by-nc-nd/2.5/).
[Puedes opinar sobre este tutorial aquí](#)



Recuerda

que el personal de [Autentia](#) te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#))

¿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?

info@autentia.com

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

Autentia = Soporte a Desarrollo & Formación

J2EE, EJBs, Struts...

[Autentia S.L.](#) Somos expertos en:
J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto

[JSP's y Modelo-Vista-Controlador](#)

[Introducción a Spring Web Flow](#)

Descripción

En este tutorial os enseñamos como crear un JSP, su relación con los servlets y como crear un ejemplo MVC en Tomcat

Spring Web Flow es un módulo de extensión del framework Spring, que facilita la implementación del flujo de páginas de una aplicación web

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.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)

	¿Buscas un hospedaje de calidad con soporte JAVA?
------------------------------------------------------------------------------------	--------------------------------------------------------------

www.AdictosAlTrabajo.com Optimizado 800X600