



JUNTA DE ANDALUCÍA


CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA

Soporte migración a WebODF

WebODF

Versión: v01r00
Fecha: 07/04/2017

Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso y por escrito de la Junta de Andalucía.

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>Dirección General de Política Digital</p>	<p>WebODF</p> <p>Soporte migración a WebODF</p>
--	---	---

HOJA DE CONTROL

Título	Soporte migración a WebODF		
Entregable	WebODF		
Nombre del Fichero	WOE_OTR_Soporte_migracion_a_WebODF_v01r00.odt		
Autor	UTE		
Versión/Edición	v01r00	Fecha Versión	07/04/17
Aprobado por	CHAP	Fecha Aprobación	07/04/17
		Nº Total Páginas	7

REGISTRO DE CAMBIOS

Versión	Causa del cambio	Responsable del cambio	Área	Fecha del cambio
v01r00	Versión inicial	UTE	UTE	07/04/17

CONTROL DE DISTRIBUCIÓN

Nombre y apellidos	Cargo	Área	Nº Copias
Manuel Perera Domínguez	Jefe de Servicio	Servicio de Coordinación de Administración Electrónica	1
Francisco González Guillén	Director de Proyecto	Servicio de Coordinación de Administración Electrónica	1
Francisco Mesa Villalba	Director de Proyecto	Servicio de Coordinación de Administración Electrónica	1
Pedro José Casanova Luis	Jefe de Proyecto	UTE	1

ÍNDICE

1	INTRODUCCIÓN.....	4
1.1	Objeto.....	4
1.2	Alcance.....	4
2	WebODF.....	5
2.1	Ventajas e inconvenientes.....	5
2.2	Integración de WebODF.....	5
2.2.1	Integración de Trew@ con WebODF.....	6
2.2.2	Función javascript de carga del editor.....	6

1 INTRODUCCIÓN

1.1 Objeto

El presente documento recoge el soporte para la migración a un nuevo editor de texto alternativo a WebOffice basado en javascript que permite la edición de los documentos en base a plantillas tras el cese del soporte de los navegadores a la ejecución de applets de java.

1.2 Alcance

Este documento se encuentra dirigido a:

- Dirección de Proyecto.
- El colectivo de usuarios y desarrolladores de sistemas que usan WebOffice.

2 WebODF

Tras el cese de los navegadores a dar soporte al plugin de Java para exploradores web basados en la arquitectura de plugin de plataforma cruzada NPAPI (tecnología necesaria para los applets Java) se ha visto necesario buscar una alternativa a WebOffice.

La alternativa al uso de WebOffice es el nuevo editor WebODT.

WebODF es una librería JavaScript que facilita la inclusión de la compatibilidad ODF (Open Document Format) a su sitio web o a su aplicación de escritorio utilizando HTML y CSS.

Se puede utilizar el nuevo editor de texto tanto para visualización como en la edición de documentos.

A diferencia de WebOffice se debe tener en cuenta que la plantilla que se facilite al editor de texto WebODF debe cargarse ya con las variables sustituidas. La sustitución de variables no debe hacerse en cliente, debe realizarse antes de abrir el documento.

2.1 Ventajas e inconvenientes

El nuevo editor de texto ofrece las siguientes ventajas e inconvenientes con respecto a WebOffice:

- Ventajas
 - Las capacidades de visión son buenas.
 - Visualización de documentos, diapositivas, hojas de cálculo, anotación de documentos.
 - Mejor rendimiento que WebOffice.
 - Mayor rapidez que WebOffice.
- Inconvenientes
 - Para la última versión disponible del editor de texto, WebODF v0.5.9, no se garantiza el correcto formato del contenido inicial en la edición de plantillas con tablas muy complejas.

Para combatir el único inconveniente, en los casos en los que se tengan plantillas muy complejas y que además deban ser editadas a voluntad de los usuarios, la única alternativa viable es que la edición se realice con el software ofimático del puesto del usuario y que el documento se incorpore al expediente como una tarea de incorporación de documentos externos.

2.2 Integración de WebODF

Para integrar WebODF se debe desplegar en su aplicación el directorio completo que se encuentra en el fichero zip con el nombre wodotexteditor.

Una vez realizado el despliegue del directorio wodotexteditor ya se puede hacer uso de la librería javascript que proporciona el componente.

2.2.1 Integración de Trew@ con WebODF

Para una correcta integración de Trew@ con la librería que proporciona el componente wodotexteditor se proporciona un fichero javascript (weboffice.js) que realiza la lógica para comunicarse de forma correcta con los servlet de descarga y subida de documentos de Trew@. La librería weboffice.js se encuentra en el fichero zip.

En el fichero proporcionado se puede apreciar la implementación de los métodos que a continuación se detallan:

- **openDocument.** Carga el nuevo editor WebOffice. Recibe como parámetro el identificador del documento a editar. Se realiza la carga utilizando el servlet de descarga de documentos de Trew@. Si el editor aún no existe, se crea uno nuevo. Si el editor existe, se quita el documento previamente cargado y se carga el nuevo.
- **eventBeforeUnloadWindow.** Función para controlar el evento de cierre de ventana del navegador cuando el editor no está en modo diálogo.
- **init.** Función que inicia la carga del nuevo editor.
- **onCloseEditorDialog.** Función callback que se ejecuta cuando el usuario intenta cerrar el editor.
- **resize.** Redimensiona adecuadamente el tamaño de la capa del editor conforme a la ventana del navegador.
- **showMessageDialog.** Función auxiliar para mostrar mensajes, solo con la finalidad de mejorar el aspecto de un simple alert. Requiere de JQuery.
- **createEditor.** Carga el nuevo editor. Es una función auxiliar, no se llama desde fuera.
- **save.** Función que se invocará al presionar el botón de guardar. Se utiliza el servlet de subida de documentos de Trew@ para almacenar el documento editado de forma correcta.

2.2.2 Función javascript de carga del editor

Para realizar la llamada a la carga del editor y realizar la parametrización de los datos necesarios utilizados por los métodos anteriormente comentados se necesita la creación de un fichero jsp, el cual será utilizado para abrir el editor. Se proporciona el fichero popup.jsp a modo ejemplo en el fichero zip.

En primer lugar en el fichero popup.jsp que se proporciona como ejemplo se puede apreciar la carga de las librerías wodotexteditor.js y weboffice.js, adicionalmente también se cargan las librerías de JQuery.

Posteriormente se encuentra el método javascript para iniciar el nuevo editor una vez cargada la página HTML.

Comentar que se deben precargar de forma correcta los siguientes parámetros:

- **urlBaseDownload:** Url del servlet de descarga de documentos
- **urlBaseUpload:** Url del servlet de subida de documentos.
- **NombreFichero:** Nombre del fichero.

Y se debe indicar el identificador del documento como parámetro de entrada al método 'openDocument'.

Para hacer uso de este fichero jsp se debe realizar la llamada desde la zona de la aplicación donde se requiera abrir el editor de la siguiente forma:

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>Dirección General de Política Digital</p>	<p>WebODF</p> <p>Soporte migración a WebODF</p>
---	---	---

```
<a href="javascript:abrirVentanaPopUpEditorDocumentos('../popup.jsp?docExp=${
documentos.documento.refDocumento}&urlServer=<%=urlServer%>&nombreFichero=${
documentos.documento.nombreFichero}', 'wofi');"
onclick="javascript:ejecutarEventoCierre=false;">
```

Para ello también es necesario definir la función javascript *abrirVentanaPopUpEditorDocumentos*:

```
function abrirVentanaPopUpEditorDocumentos(direccion, titulo) {
var popUp = window.open(direccion,
    titulo, 'Location=no,menubar=no,status=no,toolbar=no,resizable=yes,scrollbars=yes');
}
```

Se proporciona una captura de pantalla del editor WebODF con la carga de un documento de Trew@ de ejemplo en el fichero zip.