

2017



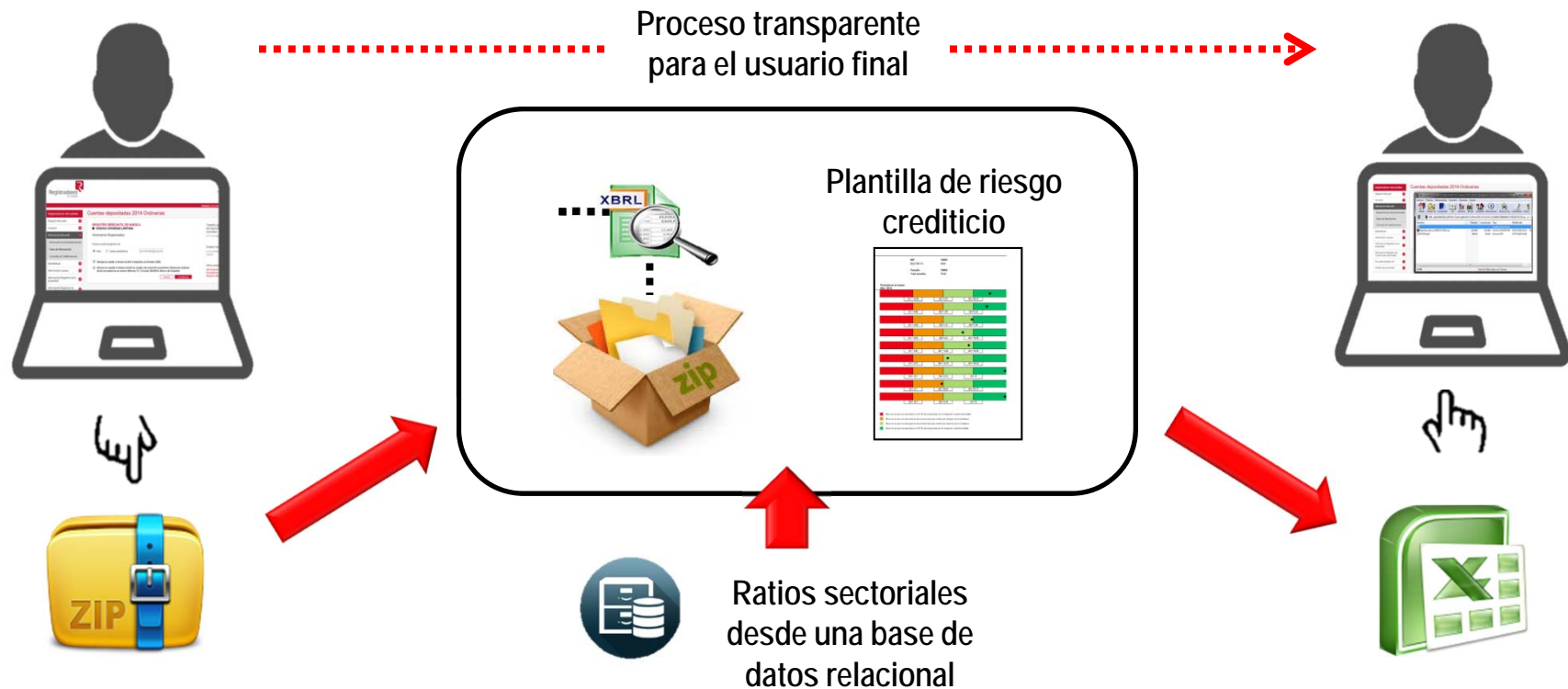
Servicio Web para plantilla de riesgo crediticio

Javier Mora González

javier.mora@xbrl.es

Asociación XBRL España

Servicio Web para plantilla de riesgo crediticio





Servicio Web para plantilla de riesgo crediticio

La Asociación XBRL España ha creado un servicio web basado en Java para la obtención automática de la plantilla de riesgo crediticio. Este proyecto es una prueba de concepto exclusivamente, con datos que no se actualizan.

[Servicio web](#) – [Informes de prueba](#) – [Código fuente Java](#) – [Diagrama funcional](#)

<http://formacionxbri.es>



UploadServlet.java

```
1  /**
2   * Copyright: Asociación XBRL España (info@xbrl.es)
3   * Este código fuente ha sido desarrollado bajo licencia CC BY (Creative Commons Attribution 4.0 International License)
4   * Más información sobre esta licencia en: http://creativecommons.org/licenses/by/4.0/
5   */
```

```
/**
 * -----
 * 1. Subir ZIP a directorio temporal
 * -----
 * Copia el fichero ZIP que hemos subido a la carpeta temporal definida en la ruta "filePath".
 * Antes de realizar la copia y para evitar duplicados, se borran todos los archivos de la carpeta temporal.
 * Se admiten ficheros ZIP hasta de un tamaño máximo de "maxFileSize".
 * Sólo se aceptan ficheros ZIP con extensión ".zip" o ".ZIP".
 */
subirZIPdirectorioTemporal(request, response);
```



```
/*
 * -----
 * 2. Descomprimir fichero ZIP
 * -----
 * Descomprime el fichero ZIP que se encuentra en la ruta "filePath", extrayendo solamente el fichero XBRL.
 * Solo puede haber un fichero XBRL dentro del fichero ZIP.
 * El archivo XBRL solo puede tener extensión ".xbrl" o ".XBRL".
 *
 * @return Fichero XBRL.
 */
File ficheroXBRL = descomprimirZIP(response);
```

```
/*
 * -----
 * 3. Leer XBRL
 * -----
 * Leemos las partidas del informe XBRL necesarias para rellenar la hoja "Carga de datos" de la plantilla Excel.
 *
 * @return Array con los valores de las partidas
 */
try {
    cargaDatos = leerXBRL(ficheroXBRL, response);
} catch (ParserConfigurationException e1) {
    // TODO Auto-generated catch block
    e1.printStackTrace();
} catch (SAXException e1) {
    // TODO Auto-generated catch block
    e1.printStackTrace();
}
```



```
/*  
* -----  
* 4. Calcular ratios  
* -----  
* Calculamos los ratios:  
*  
* T1. Tasa de variación de la cifra neta de negocios  
* R05. Resultado económico neto / Cifra neta de negocios  
* R10. Resultado económico neto / Total activo  
* R12. Resultado después de impuestos / Fondos propios  
* R21. Activos financieros a corto plazo y dispon. / Total activo  
* R20. Capital circulante / Cifra neta de negocios  
* R24. Deudas con entidades de crédito / Total patrimonio neto y pasivo  
* R22. Fondos propios / Total patrimonio neto y pasivo  
* R06. Gastos financieros / Resultado económico bruto  
*  
*/  
calcularRatios(response);
```

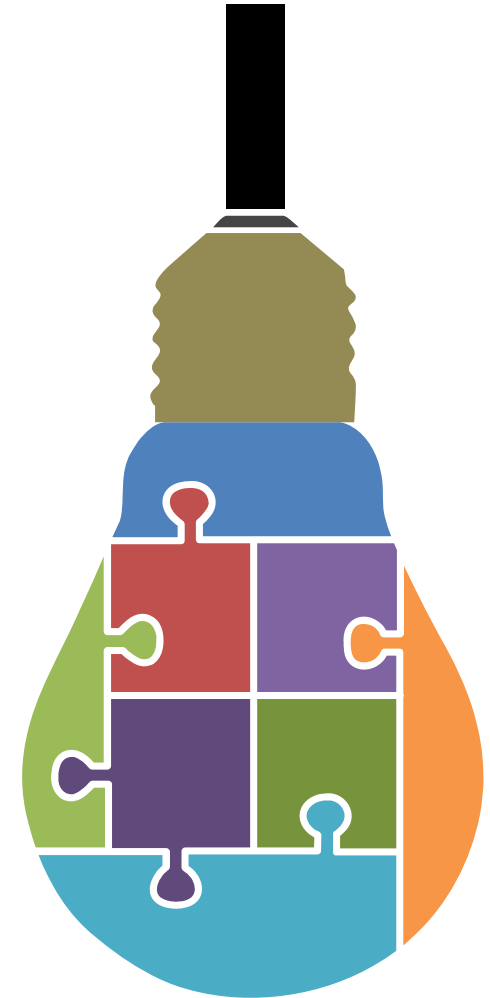


```
/*
 * -----
 * 5. Leer RSE
 * -----
 * Leemos la información de los cuartiles, así como otros datos, para la CNAE de la entidad.
 *
 */
try {
    leerRSE(response);
} catch (NamingException e) {
    e.printStackTrace();
}
```

```
/*
 * -----
 * 6. Guardar datos XBRL en Excel
 * -----
 * Guardamos los datos de identificación de la entidad así como los ratios y la información RSE en la hoja "Informe" del Excel.
 *
 */
guardarDatosExcel(response);
```

- Detalles técnicos:

- Desarrollo Java. 1270 líneas de código. JRE 1.8.0
- Página web JSP + Java Servlet
- Funcionando en un servidor Apache Tomcat 8.0
- Leyendo instancias XBRL mediante DOM
(no hay necesidad de utilizar SAX debido al reducido tamaño de los informes)
- Creación de documentos Excel con Apache POI
- Bases de datos MySQL





Demo



Gracias por su atención, ¿preguntas?



Javier Mora González

javier.mora@xbrl.es

Asociación XBRL España