

HERRAMIENTAS AVANZADAS DE SOFTWARE

LABORATORIO 5: XML en .NET

Tomando como referencia las *tareas genéricas* del laboratorio 4 vamos a extender la aplicación permitiendo a los *profesores* la importación y exportación de dichas tareas de la siguiente manera:

Importar: Las tareas genéricas contenidas en el fichero **CodAsig.xml**, se añadirán a la tabla **TareasGenericas** de la BD.

Exportar: Con las tareas genéricas de una asignatura (**CodAsig**) se genera el fichero **CodAsig.xml**

Los ficheros **.xml** tendrán como nombre el código de la asignatura (HAS.xml, SEG.xml, ...) y serán válidos conforme al esquema **tareas.xsd** que se muestra a continuación:

```
<?xml version="1.0" encoding="Windows-1252"?>
<xs:schema xmlns:has="http://ji.ehu.es/has" attributeFormDefault="unqualified"
elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="tareas">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" name="tarea">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="descripcion" type="xs:string" />
              <xs:element name="hestimadas" type="xs:unsignedByte" />
              <xs:element name="explotacion" type="xs:boolean" />
              <xs:element name="tipotarea" type="xs:string" />
            </xs:sequence>
            <xs:attribute name="codigo" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

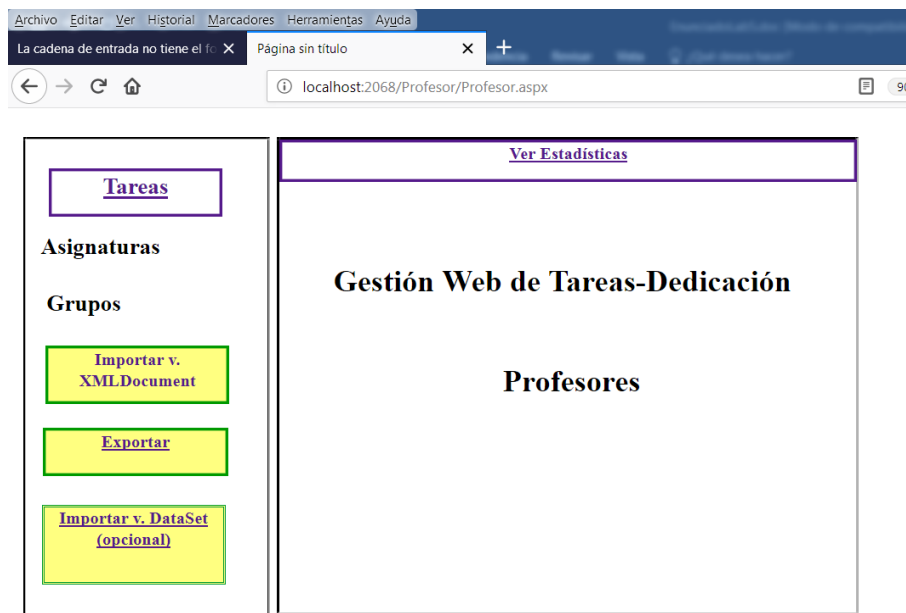
El fichero **HAS.xml** tiene el siguiente contenido:

```
<?xml version="1.0" standalone="yes" ?>
<tareas xmlns:has="http://ji.ehu.es/has">
  <tarea codigo= HAS20Gxx-Test1>
    <descripcion>Test 1 de la asignatura</descripcion>
    <hestimadas>1</hestimadas>
    <explotacion>false</explotacion>
    <tipotarea>Examen</tipotarea>
  </tarea>
  <tarea codigo=HAS20Gxx-Lab4</codigo>
    <descripcion>Laboratorio 4 ADO.NET</descripcion>
    <hestimadas>12</hestimadas>
    <explotacion>true</explotacion>
    <tipotarea>Laboratorio</tipotarea>
  </tarea>
</tareas>
```

Cada grupo deberá cambiar Gxx por el número de su grupo ya que sino solo las podrá importar un grupo. Deberéis hacer lo mismo con las tareas del fichero SEG.xml

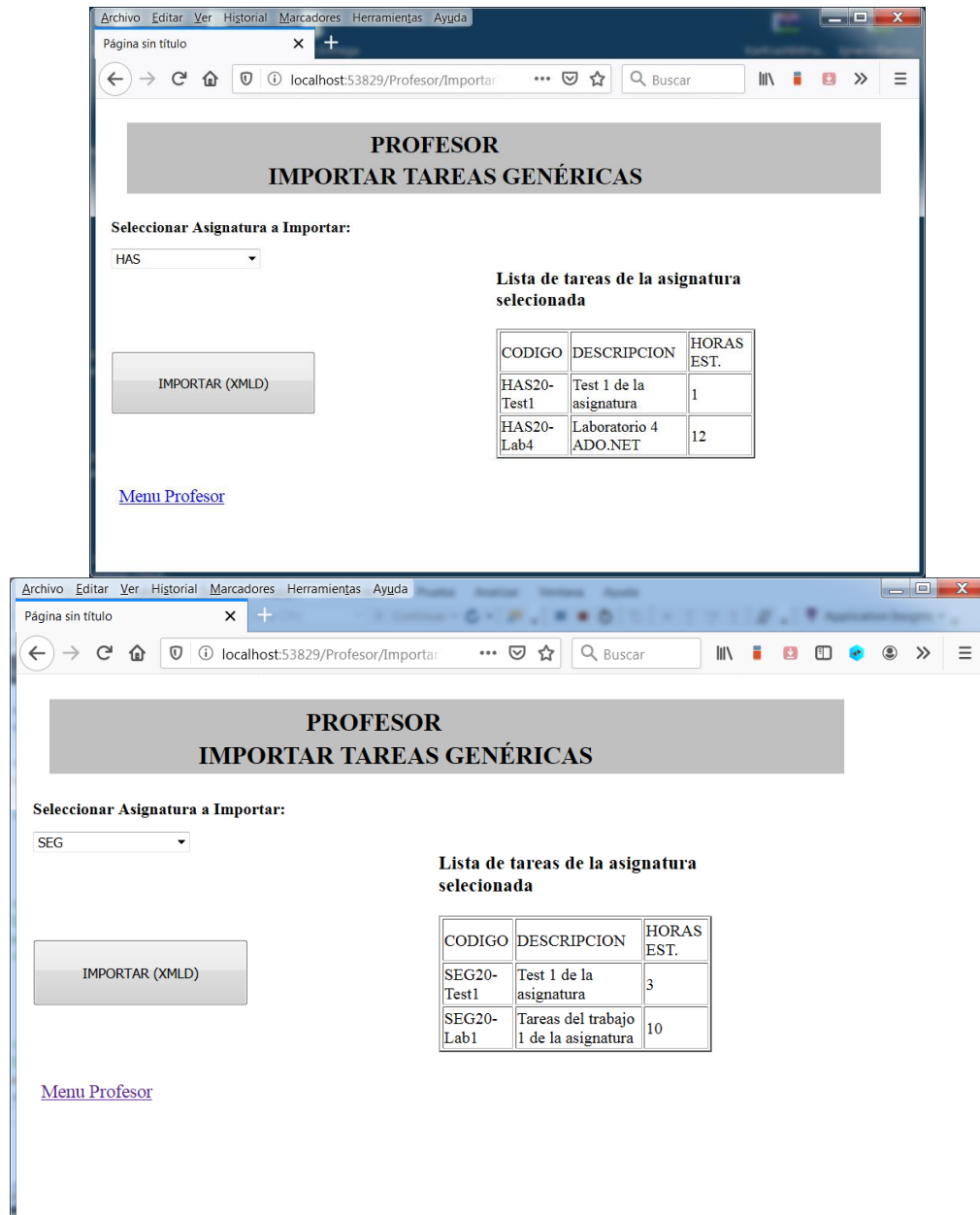
TAREAS BÁSICAS

1. Añadir a vuestro proyecto web los ficheros XML, XSD y XSL dados. Se recomienda añadir estos ficheros a una carpeta de nombre **App_Data**. Probar el editor de XML que incluye VS.NET abriendo estos ficheros.
2. Modificar el menú del profesor para incorporar tres nuevos casos de uso (dos básicos y uno opcional).



a) Importar tareas (versión XMLDocument)

Muestra al profesor una lista desplegable con las asignaturas que imparte, al seleccionar una asignatura se muestra el contenido del documento XML asociado a dicha asignatura utilizando para ello la transformación XSL contenida en *XSLTFile.xsl*.



Para aplicar la transformación se usará un control XML (caja de herramientas controles estándar). En tiempo de ejecución se deberán asignar adecuadamente las propiedades. Por ejemplo, para visualizar los datos de HAS.xml se deberá incluir el siguiente código:

```
Xml1.DocumentSource = Server.MapPath("App_Data/" &
                                   DropDownList1.SelectedValue & ".xml")
Xml1.TransformSource = Server.MapPath("App_Data/XSLTFile.xsl")
```

Para implementar la funcionalidad de importación se deberá usar **OBLIGATORIAMENTE** la clase **XMLDocument**. Cargar el documento XML contenido en el fichero con extensión .xml en una instancia de la clase **XMLDocument**. Generar un **DataTable** con la tabla **TareasGenericas** de la BD y añadir las tareas del fichero XML a la tabla del **DataSet** creado. Finalmente se salvará el **DataSet** en la BD. Con esto quedarán importadas las tareas del documento .xml en la BD.

b) Exportar tareas (DataSet)

Similar a importar con la diferencia de que se salvará en un fichero XML, de nombre igual al código de la asignatura seleccionada, el conjunto de tareas genéricas que contiene la tabla de la BD. Utilizar un **DataSet** y un **GridView** para visualizar las tareas a exportar.

codigo	descripcion	hestimadas	explotacion	tipotarea
ISO20-Lab1	Lab1	12	<input checked="" type="checkbox"/>	Laboratorio
ISO20-Test2	Segundo Examen test	1	<input checked="" type="checkbox"/>	Examen

Si se exportan las tareas para la asignatura ISO (para ello se debe hacer login con goni@ehu.es), se generará un fichero de nombre **ISO.xml** con un contenido similar a este.

```
<?xml version="1.0" standalone="yes"?>
<tareas xmlns:has="http://ji.ehu.es/has">
  <tarea codigo="ISO20-Lab1">
    <descripcion>Lab 1</descripcion>
    <hestimadas>12</hestimadas>
    <explotacion>true</explotacion>
    <tipotarea>Laboratorio</tipotarea>
  </tarea>
  <tarea codigo="ISO20-Test2">
    <descripcion>Segundo Examen test</descripcion>
    <hestimadas>2</hestimadas>
    <explotacion>true</explotacion>
    <tipotarea>Examen</tipotarea>
  </tarea>
</tareas>
```

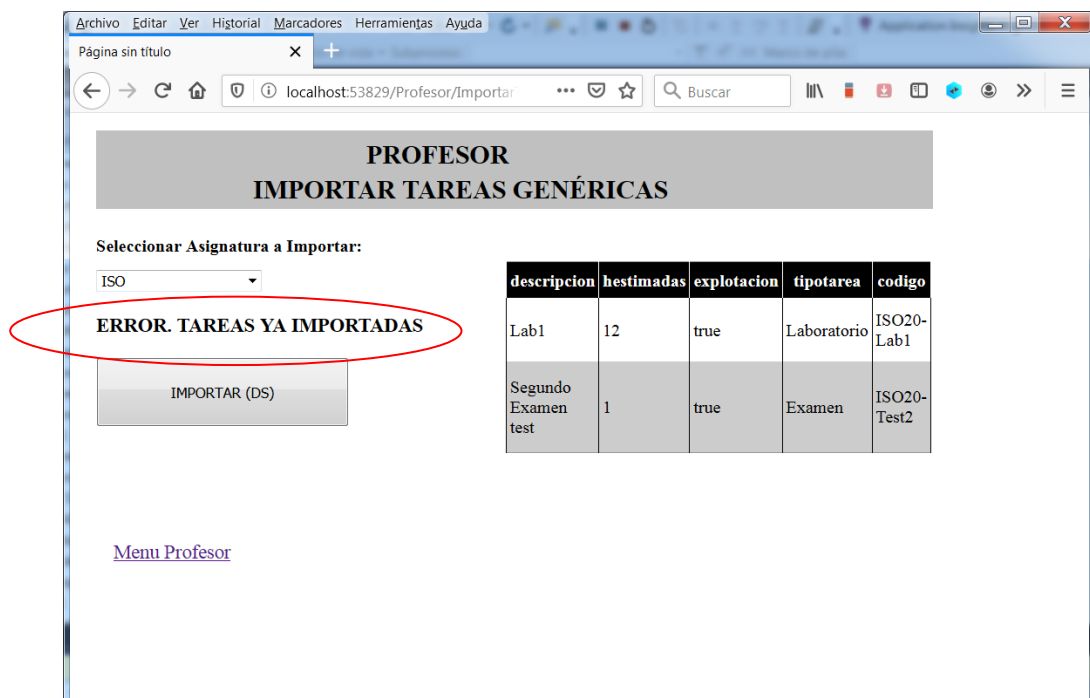
Ayuda: por defecto cuando se crea un fichero *xml* a partir de un **DataTable**, todas las columnas se convierten en Elementos (por ejemplo <descripcion>). Si queremos que una columna sea otro tipo de nodo hay que utilizar la propiedad **ColumnMapping**.

TAREAS OPCIONALES

1. Importar tareas (versión *DataSet*)

Similar a la versión obligatoria, pero sin utilizar la clase `XMLDocument`. Para cargar el fichero XML en memoria y procesarlo se utilizará un ***DataTable*** de un ***DataSet***. Tened en cuenta que la estructura del fichero XML y la de la tabla no es la misma, será necesario modificar algunas propiedades del `DataTable` antes de invocar el método `Update`.

Si las tareas ya están importadas en la BD se avisará al profesor indicando tal situación. En la siguiente captura de pantalla se ve como se intenta importar las tareas exportadas al fichero `ISO.xml`. Lógicamente si las hemos exportado es que se encuentran en la BD, por ello nos avisa de la situación errónea. (podéis hacer las pruebas con el usuario ***goni@ehu.es***)



2. Exportar tareas añadiendo el atributo `xmlns:has`

Extender la tarea obligatoria 2-b de tal forma que el fichero exportado incluya en la etiqueta `<tarea>` el atributo: `xmlns:has="http://ji.ehu.es/has"`

Una vez salvado el fichero xml, cargarlo y modificarlo mediante los métodos de la clase `XMLDocument`.

```
<?xml version="1.0" standalone="yes"?>
<tareas xmlns:has="http://ji.ehu.es/has">
.....
</tareas>
```

3. Transformaciones XSLT

Utilizando transformaciones XSL, extender la funcionalidad de la tarea obligatoria 2-a), de tal forma que la visualización de las tareas de una asignatura las presente ordenadas por el campo que desee el usuario (considerar sólo los tres campos iniciales: *codigo*, *descripcion* y *hestimadas*).

4. Exportar tareas genéricas en formato JSON

Ampliar la funcionalidad de exportar tareas, añadiendo un nuevo botón de comando (EXPORTAR JSON). En este caso se deberá generar un fichero con extensión .json (ISO.json)

```
[
  {
    "codigo": "ISO-Lab1",
    "descripcion": "Lab1",
    "hestimadas": 12,
    "explotacion": true,
    "tipotarea": "Laboratorio"
  },
  {
    "codigo": "ISO-Test2",
    "descripcion": "Segundo examen test",
    "hestimadas": 1,
    "explotacion": true,
    "tipotarea": "Examen"
  }
]
```

Hay varias formas de obtener el código json. Una de ellas es mediante la librería *Newtonsoft.Json*. Esta librería permite serializar objetos de tipo *DataTable*. No viene por defecto con Visual Studio, pero se puede añadir al proyecto como un paquete NuGet:

(<http://es.stackoverflow.com/questions/1162/c%C3%B3mo-agregar-el-paquete-de-nuget-json-net-en-un-proyecto-de-visual-studio>)