

1.Creación de la capa de negocios

La capa del Dominio de Negocios consta de una clase creada a partir de la tabla 'Libro' a la que le he añadido los atributos no persistentes: 'CodPais', 'artic' y 'digControl'.

He decidido que sean de tipo Integer para tener que despreocuparme del hecho de que al declararlos String habría que comprobar que fuesen números. Por otro lado, esta decisión hará que, seguramente en algún momento, para tratarlos tendré que usar el método toString().



También añadido a la clase 'Libro' los siguientes métodos: (que programaré más adelante)

- private void construirISBN(String pais, Integer artic, Integer digito)
Primero invocará el otro método para comprobar si el ISBN es correcto y si es así lo construirá, y si no lo es, mandará un mensaje de error.
- private boolean comprobarISBN(String ISBN)
Comprobará según el anexo al examen si el ISBN es correcto.

He decidido que el ISBN se construya en el momento que el usuario haya rellenado todos los campos necesarios. Para ello debo modificar los métodos siguientes:

- public void setArtic(Integer value) {
 setAttributeInternal(ARTIC, value);

 //Si se han introducido previamente el código del pais y el dígito de control
 //entonces se crea el ISBN:

 if (getCodPais()!=null && getDigControl()!=null)
 construirISBN(getCodPais(), value, getDigControl());
}
- public void setDigControl(Integer value) {
 setAttributeInternal(DIGCONTROL, value);

 //Si se han introducido previamente el código del pais y el articulo
 //entonces se crea el ISBN:

 if (getArtic()!=null && getCodPais()!=null) construirISBN(getCodPais(),
getArtic(), value);

```

}

• public void setCodPais(Integer value) {
    setAttributeInternal(CODPAIS, value);

    //Si se han introducido previamente el articulo y el dígito de control
    //entonces se crea el ISBN:

    if (getArtic()!=null && getDigControl()!=null) construirISBN(value, getArtic(),
getDigControl());
}

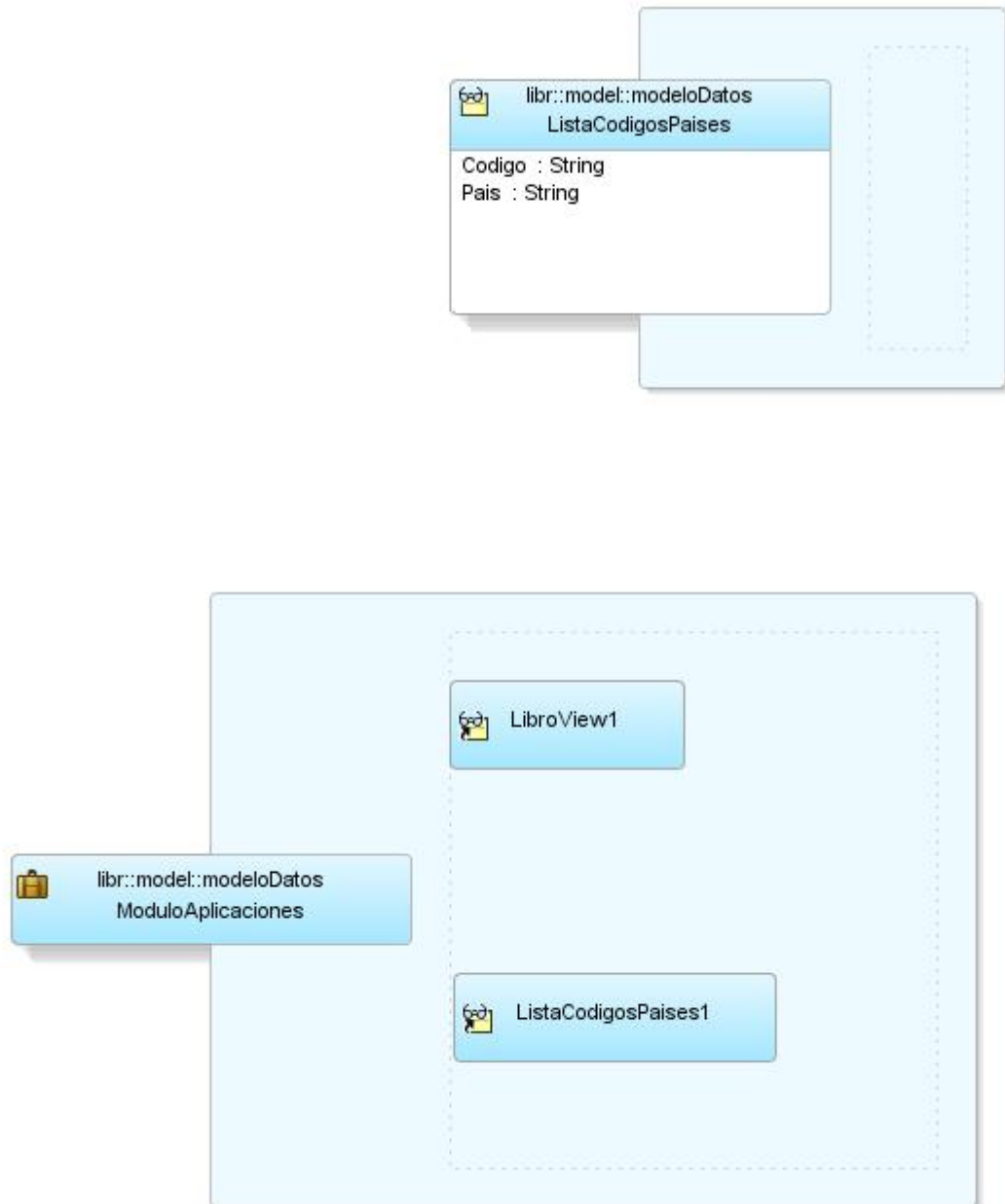
```

2.Creación del modelo de datos

A partir del modelo de datos que se genera automáticamente de la capa de negocios, hago una modificaciones.

Primero creo el Diagrama del Modelo de Datos, al que le añado un objeto vista 'ListaCodigosPaises'. Este objeto vista nos servirá para crear en la capa de presentación la lista desplegable de países.

Además le añado al diagrama el 'ModuloAplicaciones' , que automáticamente lleva asociado 'LibroView1', y además a éste último le añado el nuevo objeto vista, para que la lista salga en nuestra aplicación.

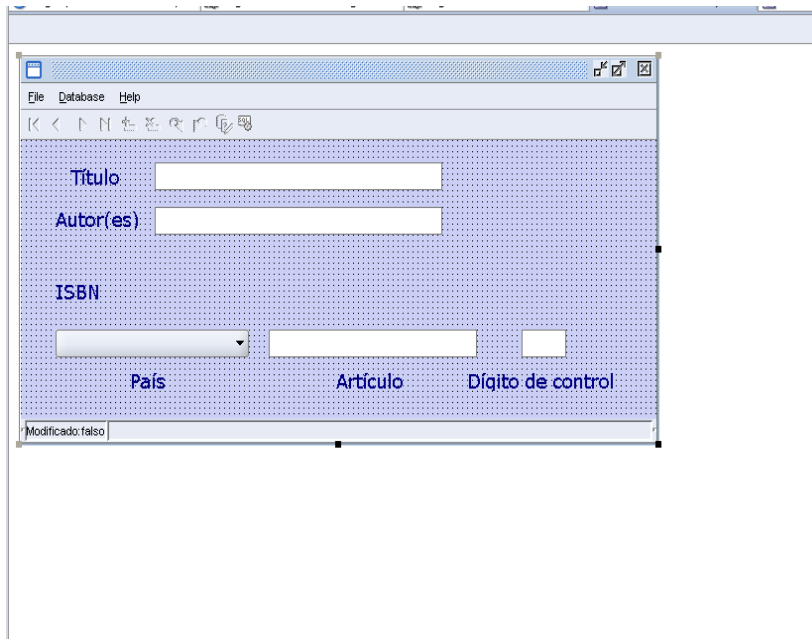


3.Creación de la capa de presentación

Creo un nuevo formulario y le añado los elementos necesarios para que tenga el aspecto necesario.

Durante la creación de la lista dinámica he tenido que cambiar el tipo de dato de CodPais a String, para que no me diera problemas.

El aspecto final del formulario es:



3. Programación de los métodos

Los métodos que he indicado antes quedan finalmente:

```

private void construirISBN(String pais, Integer artic, Integer digito){
    String ISBN;
    ISBN=978+pais+artic+digito;
    //Más adelante, cuando transformemos a Integer se comprobara que son
    numeros
    //ya que si hubiera alguna letra daría una excepción
    //De todas manera en el único sitio donde podría haber problemas es en
    pais
    //y éste viene dado por la tabla CodigoPais, que cabe esperar esté bien
    construida
    if (comprobarISBN(ISBN)){
        setISBN(ISBN);
        System.out.println("El ISBN es:"+getISBN());
    }
    else throw new JboException("Los datos proporcionados para el ISBN son
    erróneos.");
}

private boolean comprobarISBN(String ISBN){
    //Compruebo que el ISBN tiene exactamente 13 dígitos:
    if (ISBN.length()!=13) return false;
    //Elimino el dígito de control
    String dato=ISBN.substring(0,12);
    //Sumo los dígitos de lugares pares con los de impares
    Integer par=0;
    Integer impar=0;
    Integer resto=0;
    par=Integer.parseInt(dato.substring(1,2))

```

```

+Integer.parseInt(dato.substring(3,4))
+Integer.parseInt(dato.substring(5,6))
+Integer.parseInt(dato.substring(7,8))
+Integer.parseInt(dato.substring(9,10))
+Integer.parseInt(dato.substring(11,12));
impar=Integer.parseInt(dato.substring(0,1))
+Integer.parseInt(dato.substring(2,3))
+Integer.parseInt(dato.substring(4,5))
+Integer.parseInt(dato.substring(6,7))
+Integer.parseInt(dato.substring(8,9))
+Integer.parseInt(dato.substring(10,11));
par=par*3;
resto=(impar+par)%10;

resto=10-resto;
if (resto==10) resto=0;

System.out.println (ISBN+ " " + par + " " + impar + " " + resto);
if(! ISBN.substring(12).equals(resto.toString())) return false;
return true;
}

```

3.Pregunta extra

Para resolver he creado un método que rellena los campos de los atributos transitorios:

```

private void rellenaCampos()
{
String value = getIsbn();
System.out.println ("rellenaCampos:"+value);
if(getCodPais()==null || getArtic()==null || getDigControl()==null){
Integer num= new Integer(8);
if (value.substring(3,4).compareTo(num.toString())==0){
System.out.println (value.substring(3,5)+ " "+value.substring(5,12) );
setCodPais(value.substring(3,5));
setArtic(Integer.parseInt(value.substring(5,12)));
}else{
System.out.println (value.substring(3,4)+ " "+value.substring(4,12) );
setCodPais(value.substring(3,4));
setArtic(Integer.parseInt(value.substring(4,12)));
}
System.out.println (value.substring(12));
setDigControl(Integer.parseInt(value.substring(12)));
}}

```

Este método se invoca cuando se modifica el contenido de la caja del título del libro:

```

public String getTitul() {
if (getIsbn()!=null) rellenaCampos();
return (String)getAttributeInternal(TITUL);
}

```