

Introducción

En este documento se presenta la primera parte de un proyecto que consiste en reimplementar un proyecto inicial, utilizando los frameworks JSF e Hibernate. En dicho proyecto se han desarrollado dos casos de uso que permiten consultar y establecer ofertas de casas rurales (Query availability y Set availability). Dicho proyecto ya existente ha sido implementado usando la siguiente tecnología: Java Swing, servicios Web JAX-WS y el SGBD orientado a objetos objectDB.

En esta primera parte se pretende sustituir la capa de presentación que se proporciona (construida con Java Swing) por una nueva realizada utilizando el framework JSF. En esta nueva implementación dicha capa utilizará la lógica del negocio existente implementada con objectDB.

Pasos a seguir

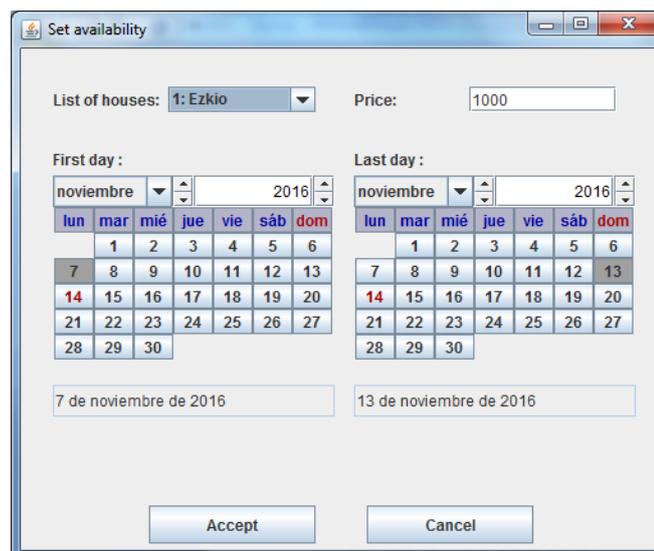
1. Instalar la aplicación pre-existente

Hay que descargar el código Java del proyecto WS-RuralHouse. Este código se encuentra disponible en el curso, en el apartado “Desarrollo del Proyecto”

Importar el proyecto en el Spring Tool Suite

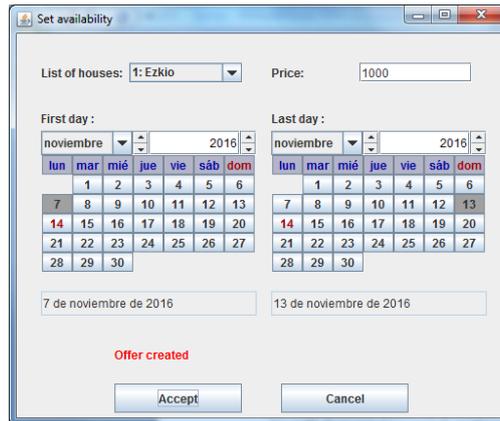
Ejecutar la aplicación. Se ejecuta haciendo “Run gui.ApplicationLauncher”.

Para comprobar su funcionamiento se puede ejecutar el caso de uso “Set Availability” e introducir una oferta para la casa “1:Ezkio” desde el 7 de noviembre al 13 de noviembre de 2016 por un total de 1000 euros, tal y como aparece en la siguiente imagen:

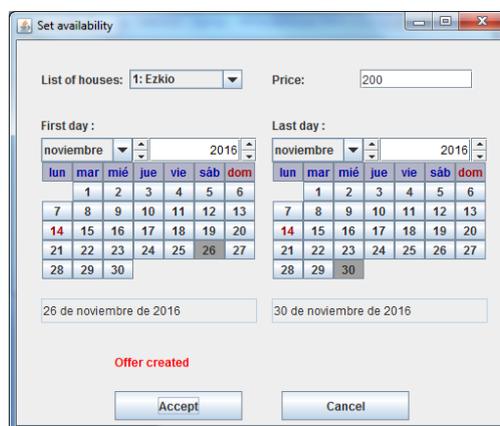


lun	mar	mié	jue	vie	sáb	dom
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

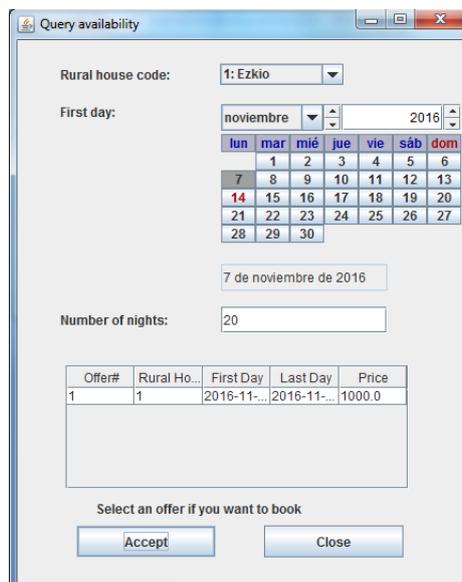
Tras pulsar el botón “Accept” aparecerá un mensaje que indica que se ha creado la oferta:



Se puede crear otra oferta entre el 26 y 30 de noviembre por un total de 200 euros.



Ejecutando el otro caso de uso, Query Availability, y preguntando por todas las ofertas de la casa "1:Ezkio" que están incluidas completamente en el intervalo de tiempo que comienza el 7 de noviembre y termina 20 días después (esto es el 27 de noviembre), entonces nos aparece la oferta 7/11/2016 a 13/11/2016. Esta oferta se muestra ya que todos esos días están comprendidos en el intervalo 7/11/2016 a 27/11/2016. Sin embargo la oferta del 26 al 30 de noviembre no se muestra porque los días 28 al 30 de noviembre de la oferta no están en el intervalo.

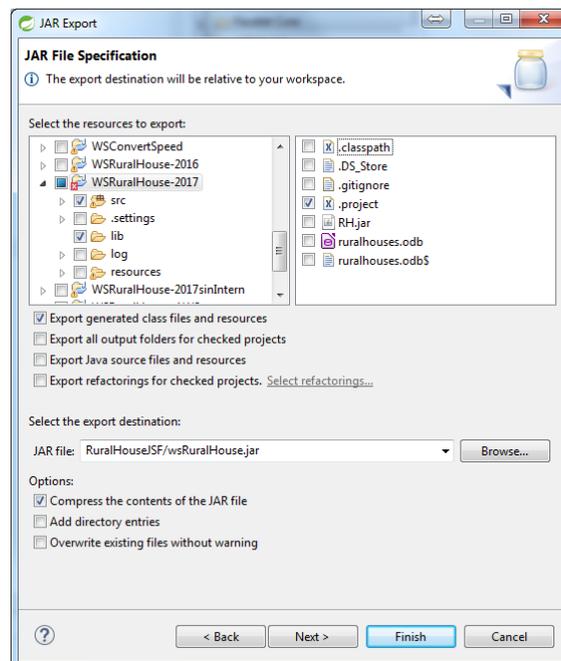


2. Crear un nuevo proyecto JSF que utilice la lógica del negocio correspondiente al proyecto pre-existente.

Después de crear la aplicación JSF (tal y como se realizó en el laboratorio de JSF), al que podemos dar el nombre RuralHouseJSF) hay que crear e incluir en este proyecto una librería (un fichero .jar) con la lógica del negocio del proyecto WS-RuralHouse con el objetivo de ser usada en este nuevo proyecto (en el que queremos reutilizar la lógica del negocio). Para ello hay que:

1) Crear la librería con el proyecto WS-RuralHouse:

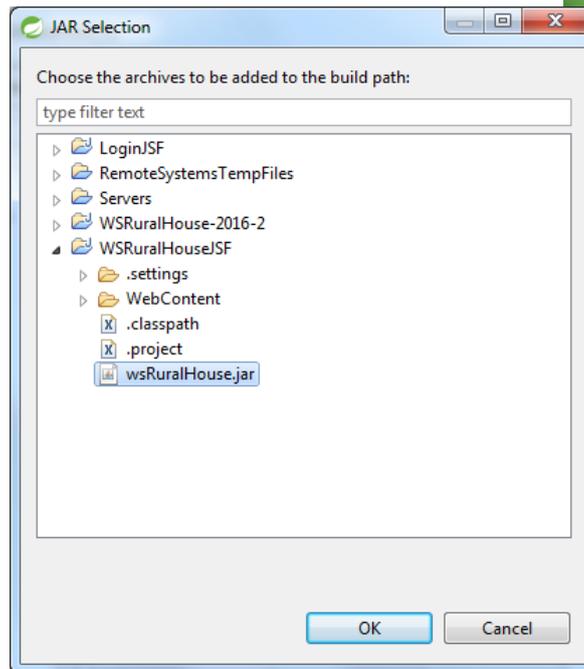
Seleccionar el proyecto WS-RuralHouse en el Project Explorer => Hacer Click Dcho. => Export => Java => Jar File => Incluir las librerías que aparecen en el directorio “lib” (que contiene las librerías de objectDB y de jCalendar) y las clases Java que están en el directorio “src” (en realidad se exportarán los .class correspondientes a esas clases, al estar activada la opción “Export generated class files and resources”) => Dar un nombre (por ejemplo wsRuralHouse.jar) y guardar el fichero en el mismo directorio donde se haya creado el nuevo proyecto RuralHouseJSF), tal y como aparece a continuación:



La librería se genera, a pesar de que avisa de la existencia de unos warning en el proyecto proporcionado, que no impiden el correcto funcionamiento de la aplicación, por lo que se mantendrán tal cual.

2) Añadir las librerías al proyecto WS-RuralHouseJSF

Project => Properties => Java Build Path => Add Jar (como se ha guardado en el mismo directorio del proyecto, entonces estará accesible en el mismo proyecto, tal y como se ve en la siguiente figura)



Nota: si se incluyen así las librerías así en vez de con “Add External Jar”, al importar posteriormente el proyecto en otro ordenador se usan nombres de librerías con sus caminos relativos y no con caminos completos, evitando problemas de librerías no encontradas.

Nota: la librería de objectdb.jar también hay que incluirla, porque aunque esté dentro de wsRuralHouse.jar, no es accesible en ese formato (las clases de objectDB no se encuentran en los directorios com\... ni javax\...)

3) Que dichas librerías estén accesibles para el servidor web que ejecutará el WS-RuralHouseJSF (esto es, que se despliegue correctamente).

Project => Properties => Deployment Asssembly => Add => Java Build Path (el lugar donde se encuentran las librerías del proyecto actual) => Next => Seleccionar la librería wsRuralHouse.jar y objectDB => Finish

A partir de este momento hay que trabajar en el proyecto: **crear las vistas JSF, beans gestionados y configuración necesaria para la nueva aplicación web.**

