

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 apuestas (*Create Question* y *Query Questions*). 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 Bets. El proyecto se encuentra en: <https://github.com/jononekin/Rides24>

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

Para comprobar su funcionamiento se puede ejecutar el caso de uso “Create Ride” e introducir un viaje para el 13 de noviembre de 2024, de Donosti a Bilbao, de 2 asientos y 10 euros, tal y como aparece en la siguiente imagen:

The screenshot shows a Java Swing window titled "Create Ride". It has a light gray background and standard window controls (minimize, maximize, close) in the top right corner. The window is divided into two main sections. On the left, there are four input fields: "Depart City" with the text "Donostia", "Arrival C..." with "Bilbao", "Number of Seats" with "2", and "Price" with "10". On the right, there is a "Ride date" section with a calendar for November 2024. The calendar shows days from Sunday to Saturday, with the 13th highlighted in red. At the bottom of the window, there are two buttons: "Create Ride" and "Close".

Tras pulsar el botón “Create Ride” aparecerá un mensaje que indica que se ha creado la pregunta:

Ingeniería del Software II
Proyecto Rides con JSF

Create Ride

Depart City

Donostia

Arrival C...

Bilbao

Number of Seats

2

Price

10

Ride date

November

2024

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					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

Ride created

Create Ride

Close

Se puede crear otro viaje para el 17 de noviembre de 2024, de Donosti a Gasteiz, de 1 asiento y 7 euros.

Create Ride

Depart City

Donostia

Arrival C...

Gasteiz

Number of Seats

1

Price

7

Ride date

November

2024

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					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

Ride created

Create Ride

Close

Ejecutando el otro caso de uso, Query Rides, y preguntando por los viajes que acabamos de crear, veremos cómo se han añadido correctamente.

Find rides

Depart City

Donostia

Arrival C...

Bilbao

Ride date

November

2024

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					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

Viajes: November 13, 2024

Driver	Seats	Price
Test driver	2.0	10.0

Close

Ingeniería del Software II

Proyecto Rides con JSF

Driver	Seats	Price
Test driver	1.0	7.0

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 RidesJSF) hay que crear e incluir en este proyecto una librería (un fichero .jar) con la lógica del negocio del proyecto Rides 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 Bets:

Seleccionar el proyecto Rides24 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 rides.jar) y guardar el fichero tal y como aparece a continuación (e.g., en la carpeta lib del proyecto):

JAR File Specification

The export destination will be relative to your workspace.

Select the resources to export:

- ☐ HolaMundoJSF
- ☐ HolaMundoJSF2
- ☐ jsf-HelloWorld
- ☐ LoginJSF
- ☒ Rides24
- ☐ RidesJSF

☒ Export generated class files and resources

☐ Export all output folders for checked projects

☐ Export Java source files and resources

☐ Export refactorings for checked projects. [Select refactorings...](#)

Select the export destination:

JAR file: RidesJSF/lib/rides.jar [Browse...](#)

Options:

☒ Compress the contents of the JAR file

☐ Add directory entries

☐ Overwrite existing files without warning

[?](#) < Back Next > **Finish** Cancel

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 la librería al repositorio maven

Bien en la línea de comandos general o bien en la de Eclipse (Window>Show View>Terminal) navegar hasta el proyecto RidesJSF e instalar la dependencia con el siguiente comando:

```
mvn install:install-file -Dfile=lib\rides.jar -DgroupId=1.0 -DartifactId=Rides24 -
Dversion=0.0.1-SNAPSHOT -Dpackaging=jar
```

```
E:\Docencia\IS2\2024.2025.Eclipse\RidesJSF>mvn install:install-file -Dfile=lib\rides.jar -DgroupId=1.0 -DartifactId=Rides24 -Dversion=0.0.1-SNAPSHOT -Dpackaging=jar
[INFO] Scanning for projects...
[INFO]
[INFO]      org.is2.frameworks:RidesJSF
[INFO]
[INFO]
[INFO]
[INFO]
[INFO]
[INFO]      maven-install-plugin:2.4:install-file @ RidesJSF
[INFO] Installing E:\Docencia\IS2\2024.2025.Eclipse\RidesJSF\lib\rides.jar to C:\Users\scpazsem\.m2\repository\1\0\Rides24\0.0.1-SNAPSHOT\Rides24-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.421 s
[INFO] Finished at: 2024-11-11T17:30:19+01:00
[INFO]
```

3) Añadir la dependencia correspondiente en el pom.xml

```
<dependency>  
  <groupId>1.0</groupId>  
  <artifactId>Rides24</artifactId>  
  <version>0.0.1-SNAPSHOT</version>  
  <scope>compile</scope>  
</dependency>
```

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.**

Calificación

La calificación del proyecto será como sigue:

- La reimplementación de los casos de uso *Query Rides* y *Create Ride* únicamente conllevará una calificación máxima de aprobado (una calificación de 6, como máximo).
- Para obtener una calificación máxima de notable (es decir, una calificación de 8 como máximo) habrá que implementar además los casos de uso de *Login* (iniciar sesión, con acceso a “Mis viajes”) y *Register* (registro). De todas formas, la implementación de estos casos de uso no garantiza dicha calificación.
- Finalmente, para optar a la calificación de sobresaliente será necesario implementar al menos otro caso de uso que implique la creación de una nueva clase de dominio.

Autores: Alfredo Goñi Sarriuren y Maider Azanza Sesé

