

# SDI2-20

---

Enol Matilla Blanco – UO237253

Raul Rodriguez Rodriguez - UOXXXXXX

## Captura de JUnit:

The screenshot shows the Eclipse IDE interface. The top menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. Below the menu is a toolbar with various icons. The left sidebar contains the Package Explorer and the JUnit view. The JUnit view shows the test results for the class `com.sdi.tests.Tests.PlantillaSDI2_Tests1617`. The test run finished after 224,142 seconds. The results show 38/38 runs, 0 errors, and 0 failures. The right pane displays the source code of `BeanUsers.java` and `BeanTasks.java`. The code includes comments in Spanish and Java code for Selenium WebDriver. The bottom status bar shows the text `<terminated> PlantillaSDI2_Tests1617 [JUnit] ?`.

Java - sdi2-20-pruebas/src/com/sdi/tests/Tests/PlantillaSDI2\_Tests1617.java - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

Finished after 224,142 seconds

Runs: 38/38 Errors: 0 Failures: 0

com.sdi.tests.Tests.PlantillaSDI2\_Tests1617 [Runner: ...]

BeanUsers.java BeanTasks.java

```
1657 driver.  
1658 get("http:  
1659 // driver.  
1660 // get("ht  
1661  
1662 // Se carg  
1663 SeleniumUt  
1664 driver.get  
1665 "h  
1666 // driver.  
1667 // equals(  
1668 }  
1669  
1670 // PR38: Inter  
1671 // no autentic  
1672 @Test  
1673 public void pr  
1674 // Tratamc  
1675 driver.get  
1676 // driver.  
1677 // get("ht  
1678  
1679 // Se carg  
1680 SeleniumUt  
1681 driver.get  
1682 "h  
1683 // driver.  
1684 // equals(  
1685 }  
1686 }
```

Problems @ Javadoc Declaration

<terminated> PlantillaSDI2\_Tests1617 [JUnit] ?

Failure Trace

## Mapa de navegación (Sacado del faces-config.xml):

```
<navigation-rule>
    <from-view-id>*</from-view-id>
    <navigation-case>
        <from-outcome>error</from-outcome>
        <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>
<navigation-rule>
    <from-view-id>/registroUsuario.xhtml</from-view-id>
    <navigation-case>
        <from-action>#{controller.atras}</from-action>
        <from-outcome>>true</from-outcome>
        <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.registrar}</from-action>
        <from-outcome>>true</from-outcome>
        <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.registrar}</from-action>
        <from-outcome>>false</from-outcome>
        <to-view-id>/registroUsuario.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>
<navigation-rule>
    <from-view-id>/index.xhtml</from-view-id>
    <navigation-case>
        <from-action>#{controller.validar}</from-action>
        <from-outcome>admin</from-outcome>
        <to-view-id>/restricted/listaUsuarios.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.validar}</from-action>
        <from-outcome>user</from-outcome>
        <to-view-id>/usuarios/listaTareas.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.registro}</from-action>
        <from-outcome>>true</from-outcome>
        <to-view-id>/registroUsuario.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>
<navigation-rule>
    <from-view-id>/restricted/listaUsuarios.xhtml</from-view-id>
    <navigation-case>
        <from-action>#{controller.cambiarEstado(vuser)}</from-action>
        <from-outcome>exito</from-outcome>
        <to-view-id>/restricted/listaUsuarios.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.eliminar(vuser)}</from-action>
        <from-outcome>exito</from-outcome>
    </navigation-case>
</navigation-rule>
```

```

        <to-view-id>/restricted/listaUsuarios.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.reiniciarBD}</from-action>
        <from-outcome>exito</from-outcome>
        <to-view-id>/restricted/listaUsuarios.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.cerrarSesion}</from-action>
        <from-outcome>>true</from-outcome>
        <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>
<navigation-rule>
    <from-view-id>/usuarios/listaTareas.xhtml</from-view-id>
    <navigation-case>
        <from-action>#{btasks.editarTarea(vtask)}</from-action>
        <from-outcome>exito</from-outcome>
        <to-view-id>/usuarios/formTarea.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.cerrarSesion}</from-action>
        <from-outcome>>true</from-outcome>
        <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>
<navigation-rule>
    <from-view-id>/usuarios/formTarea.xhtml</from-view-id>
    <navigation-case>
        <from-action>#{btasks.modificarTarea}</from-action>
        <from-outcome>exito</from-outcome>
        <to-view-id>/usuarios/listaTareas.xhtml</to-view-id>
    </navigation-case>
    <navigation-case>
        <from-action>#{controller.cerrarSesion}</from-action>
        <from-outcome>>true</from-outcome>
        <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>

```

## Aspectos técnicos:

Para la autorización se ha decidido utilizar filtros.

Hay 3 roles:

- Amin: Solo puede entrar en las páginas de la carpeta “restricted”, y las que no tienen protección.
- User: Solo puede entrar en las páginas de la carpeta “usuarios”, y las que no tienen protección.
- Anónimo: Solo puede entrar en las páginas sin protección.

Para realizar esto se han utilizado 2 filtros, uno por carpeta con acceso restringido (Admin y User).

Se han empleado etiquetas <p:messages/> en todas las páginas para poder mostrar mensajes a los usuarios, como el de registro correcto, finalización correcta de una tarea, errores en la creación de un nuevo usuario con los datos, etc.

Todas las tablas y botones que pueden ser susceptibles de cambiar sus valores están ajaxificados, ya sea mediante un update, en el caso de las tablas (Llamado desde los botones de cambio de categoría, el de filtrar finalizadas, etc), o mediante un autoUpdate en el caso de los mensajes para el usuario.

Para las condiciones de coloreado de casillas, en lugar de usar etiquetas “c:if” se ha decidido utilizar el atributo “rendered” de las clases de PrimeFaces para mantener la integridad del código y evitar más etiquetas.

El formulario para crear tareas de la parte superior de la página “listaTareas.xhtml” se ha modificado mediante el atributo “style” para alinear elementos y ajustar tamaños para hacer la vista más limpia.

En el formulario de nuevos usuarios, las contraseñas se meten mediante unos campos “p:password” que te indican la fuerza de la contraseña introducida.

La validación de Email, así como de las contraseñas, se han realizado mediante comprobación de expresiones regulares, para permitir todas las posibles combinaciones válidas. Cualquier error en alguna de las validaciones mostrará un mensaje de Warn al usuario con información sobre qué ha metido mal, personalizado para cada posible caso.