

Ejemplo: Crear Procesos

```
java.lang.Process; java.lang.Runtime.exec();...
```

Pasos generales a seguir

Vamos a ver cómo implementar una aplicación que va lanzando procesos de otra aplicación.

Los pasos generales que vamos a seguir son:

1. Crear y generar la aplicación de la que lanzaremos sus procesos.
2. Crear un nuevo proyecto que será el creador de procesos.
3. Implementar el código necesario para crear procesos.
4. Probar la ejecución del creador de procesos.
5. Posibles mejoras.

Finalizaremos con unas recomendaciones para probar el ejemplo.

Crear una aplicación para lanzar sus procesos

- Para utilizar una aplicación funcional y no tener que implementarla completa, vamos a utilizar uno de los **ejemplos que incorpora NetBeans**, “Document Editor”.
- Puedes utilizar una aplicación que tengas ya generada si lo prefieres.
- Nos interesa es tener el ejecutable de la aplicación (en el caso de Java, su **.jar**).

Procedemos de la siguiente manera:

1. Abrimos **NetBeans**.
2. En la **Página de inicio**, seleccionamos **Probar un proyecto...**
Podemos ver la Página de inicio desde menú Ayuda → Página de inicio.

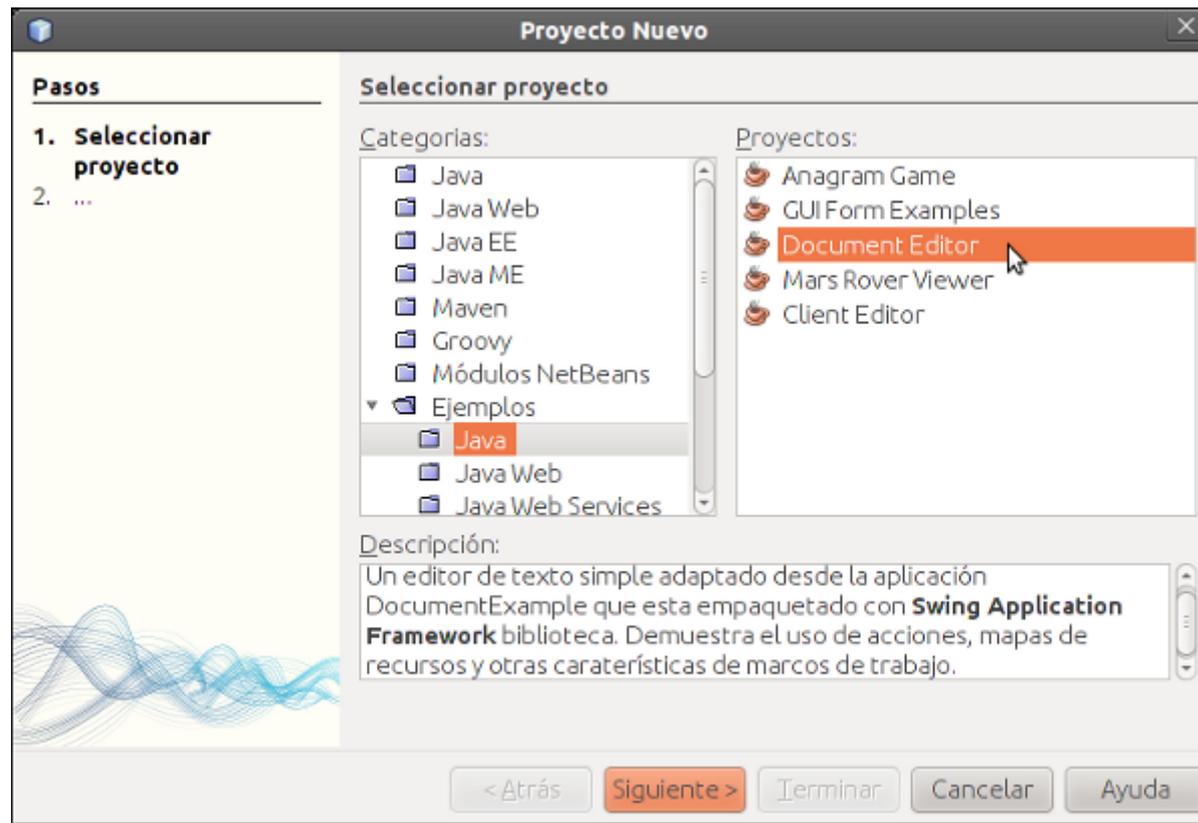


Página de inicio de NetBeans. Probar un proyecto...

Crear una aplicación para lanzar sus procesos

Continuamos:

3. El nuevo proyecto que vamos a crear es un **Ejemplo Java** llamado “**Document Editor**”. Se trata un editor de texto plano. Es una aplicación de escritorio que utiliza el Framework Swing.
4. Una vez que tengamos seleccionado lo que corresponde, pulsamos el botón **Siguiente**>.



Diálogo Proyecto Nuevo de NetBeans. Tipo de proyecto: Ejemplos → Java → Document Editor.

Crear una aplicación para lanzar sus procesos

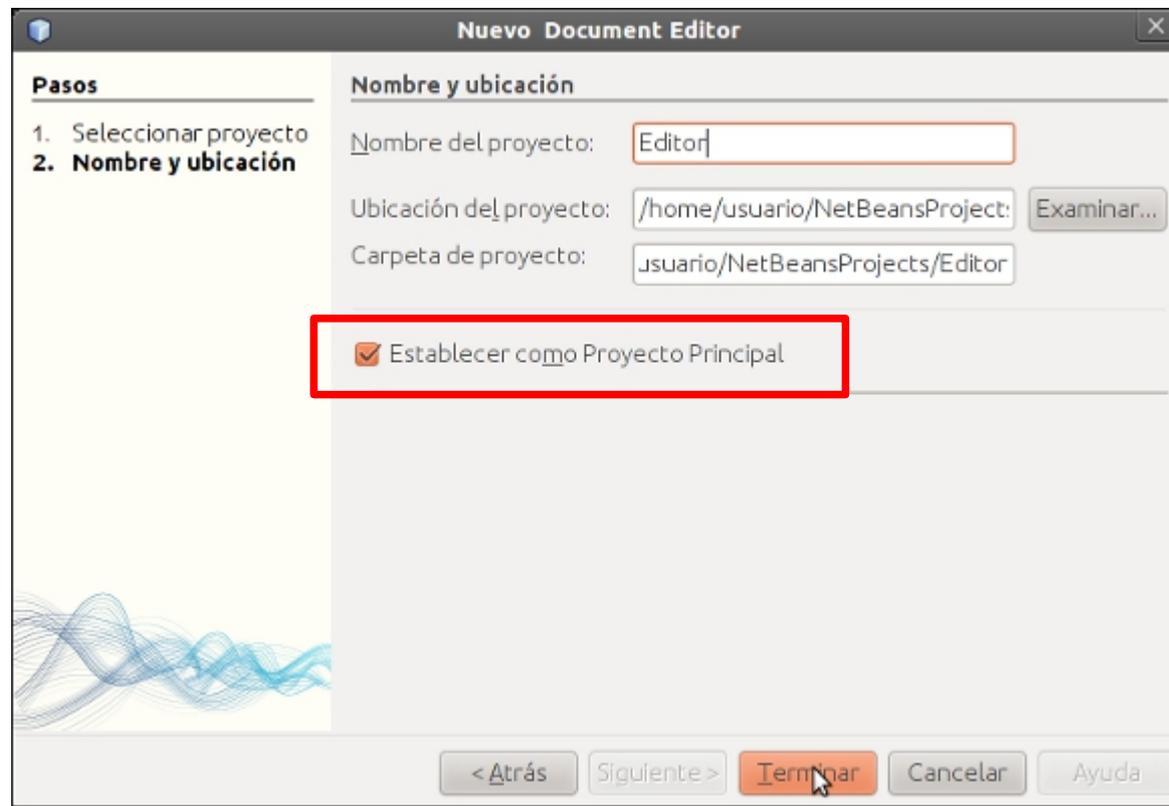
A continuación:

5. Indicamos el **nombre** con el que guardaremos el nuevo proyecto y su **ubicación**.

En mi caso, voy a llamar al proyecto “Editor” y voy a dejar que me lo guarde en la carpeta de proyectos por defecto de NetBeans.

6. Pulsamos el botón **Terminar**.

Es importante que éste proyecto se **establezca como Proyecto Principal**, para poder generar su .jar.

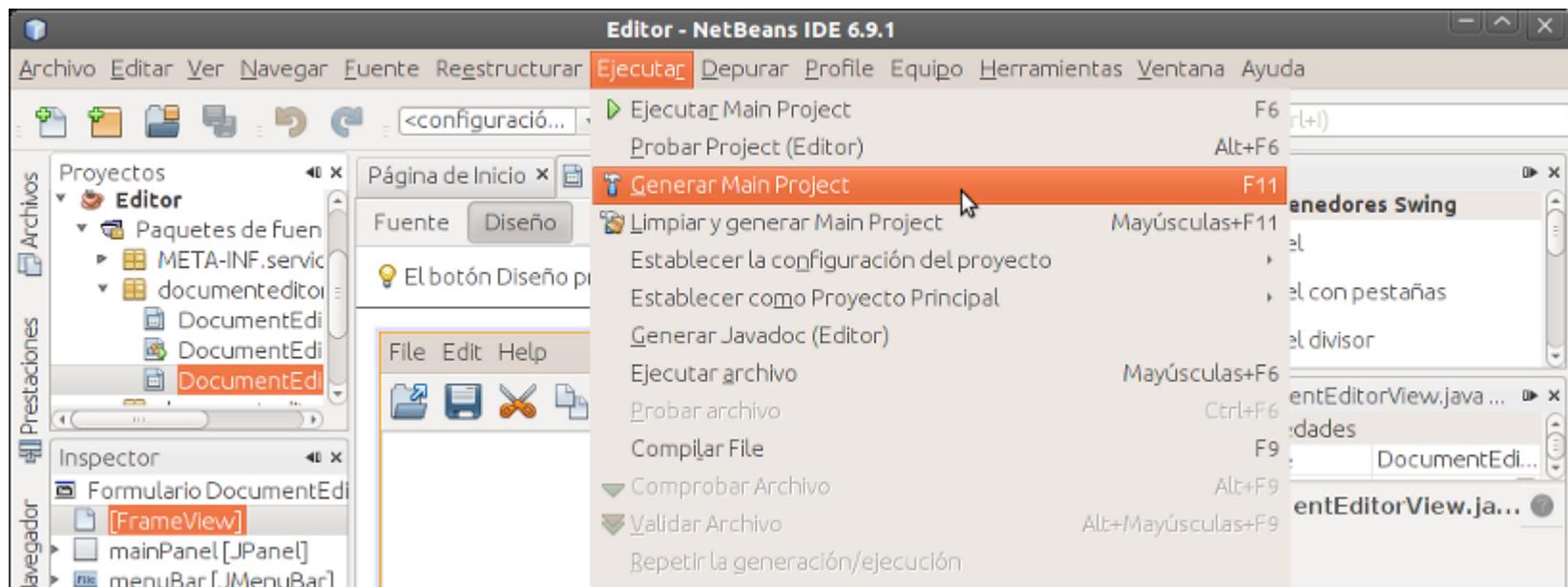


Diálogo Proyecto Nuevo de NetBeans. Nombre y ubicación del nuevo proyecto: Editor.
No olvidar marcar Establecer como Proyecto Principal si no está marcado.

Crear una aplicación para lanzar sus procesos

Sólo nos queda:

7. Generar el proyecto principal. Menú **Ejecutar** → **Generar Main Project (F11)** o Limpiar y generar Main Project (Mayús + F11).



Proyecto Editor en NetBeans. Generar proyecto principal – Main Project (F11).

La ruta en la que tendremos el ejecutable será:

- En Windows
C:\Users\usuario\Documents\NetBeansProjects\Editor\dist\DocumentEditor.jar
- En GNU/Linux
/home/usuario/NetBeansProjects/Editor/dist/DocumentEditor.jar

Crear un proyecto, el creador de procesos

- Vamos a crear un proyecto que creará procesos de tipo “Editor”.

Comencemos:

1. Añadimos un nuevo proyecto. Menú Archivo → Proyecto Nuevo... (Ctrl+Mayús+N)
2. El tipo de proyecto va a ser Java → **Java Desktop Application**. Pulsamos **Siguiente>**.



Diálogo Proyecto Nuevo de NetBeans. Tipo de proyecto: Java → Java Desktop Application.

3. El tipo de aplicación, será una aplicación Indicamos el **nombre** con el que guardaremos el nuevo proyecto y su **ubicación**.
En mi caso, voy a llamar al proyecto “CrearProcesos” y voy a dejar que me lo guarde en la carpeta de proyectos por defecto de NetBeans.

Crear un proyecto, el creador de procesos

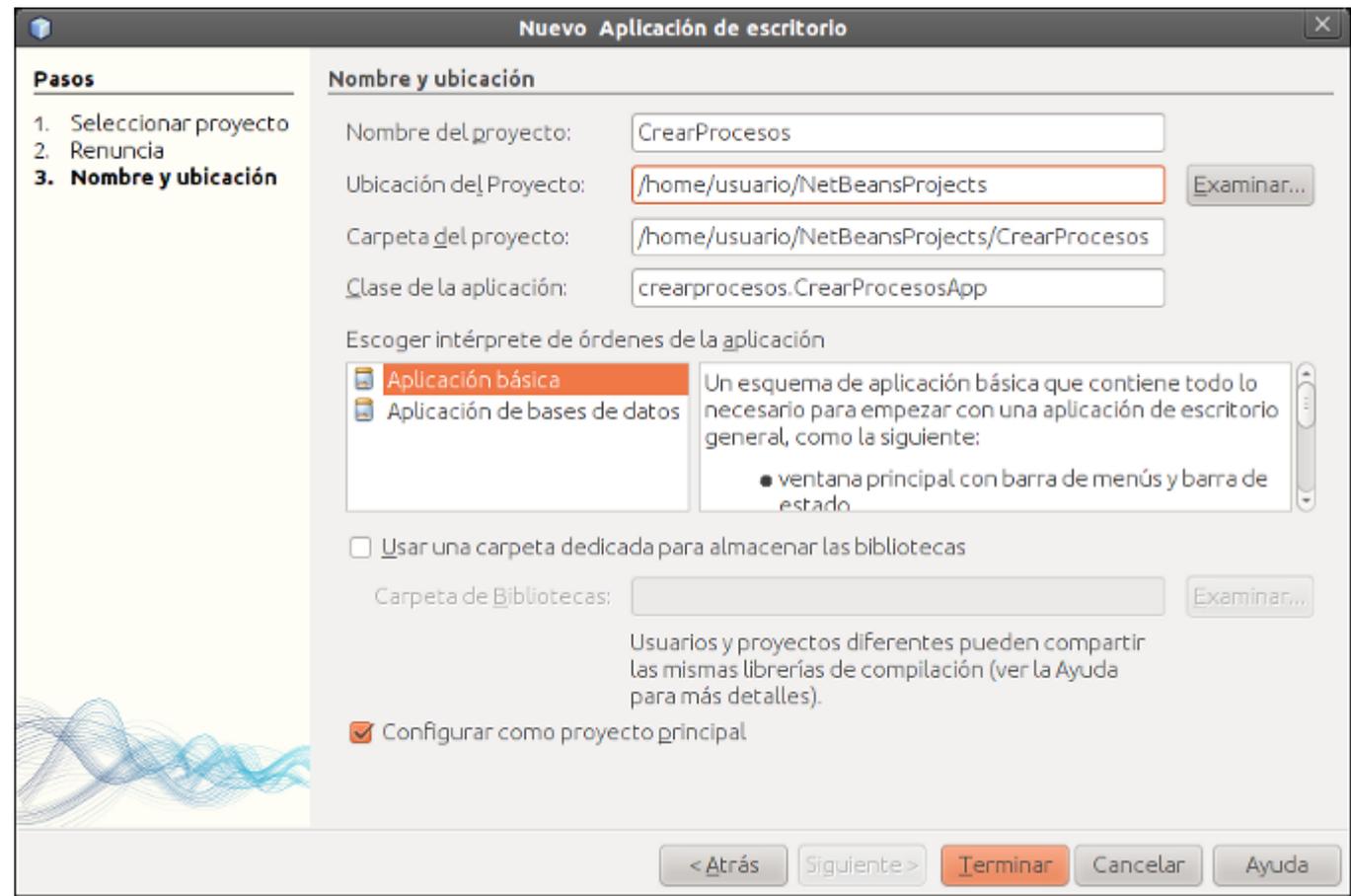
A continuación:

3. El entorno, nos muestra un mensaje en el que nos avisa de que el Framework Swing no va a seguir desarrollándose para formar parte del Kit de desarrollo java (por lo que nuestro proyecto no va a ser todo lo portable que nos gustaría). Pulsamos el botón **Siguiente>**.

4. El tipo de aplicación, será una **Aplicación básica**.

5. Indicamos el **nombre** con el que guardaremos el nuevo proyecto y su **ubicación**.

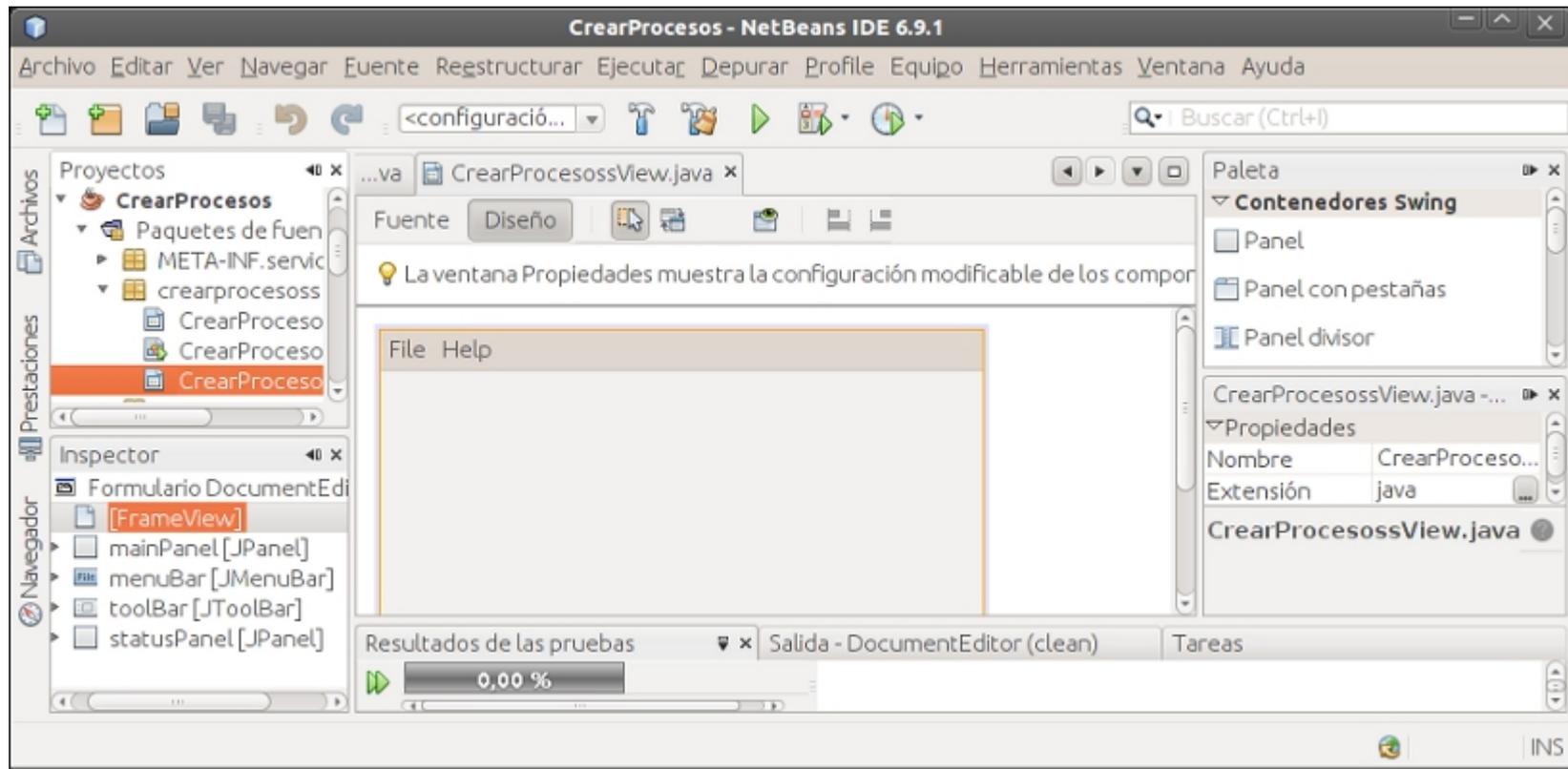
En mi caso, voy a llamar al proyecto “CrearProcesos” y voy a dejar que me lo guarde en la carpeta de proyectos por defecto de NetBeans. Una vez que tengamos seleccionado lo que corresponde, pulsamos el botón **Siguiente>**.



Diálogo Proyecto Nueva Aplicación de escritorio. Tipo: Aplicación básica.
Nombre: CrearProcesos

Crear un proyecto, el creador de procesos

Esta es nuestro proyecto de partida:



NetBeans Proyecto: CrearProcesos recién creado.

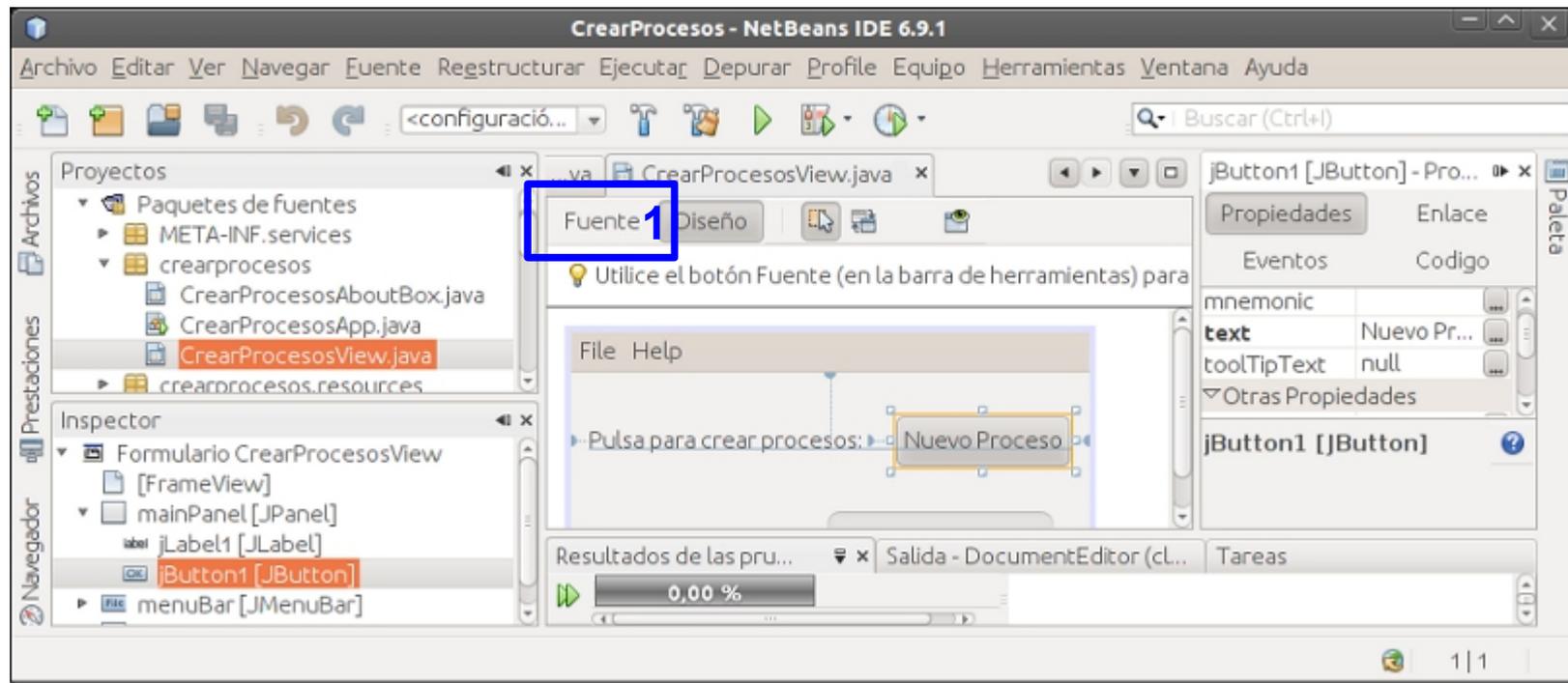
Lo siguiente que vamos a hacer es incluir un botón e implementar el código que cree nuevos procesos de la aplicación Editor.

Implementar: Crear procesos.

Comenzamos incluyendo un control de tipo botón (JButton):

- Modificamos las propiedades name y text del botón, para que el aspecto de la interfaz sea coherente.
- Además incluimos una etiqueta (JLabel), para que dé más información al usuario.

Recuerda que también debes modificar, al menos, las propiedades name y text del label. Así tendrá un nombre identificable en la aplicación y un texto coherente.



NetBeans Proyecto CrearProcesos. Incluyendo controles en la vista: botón y etiqueta.

Por fin llegamos a la implementación:

1. Nos vamos al código fuente de la vista (View).

Implementar: Crear procesos.

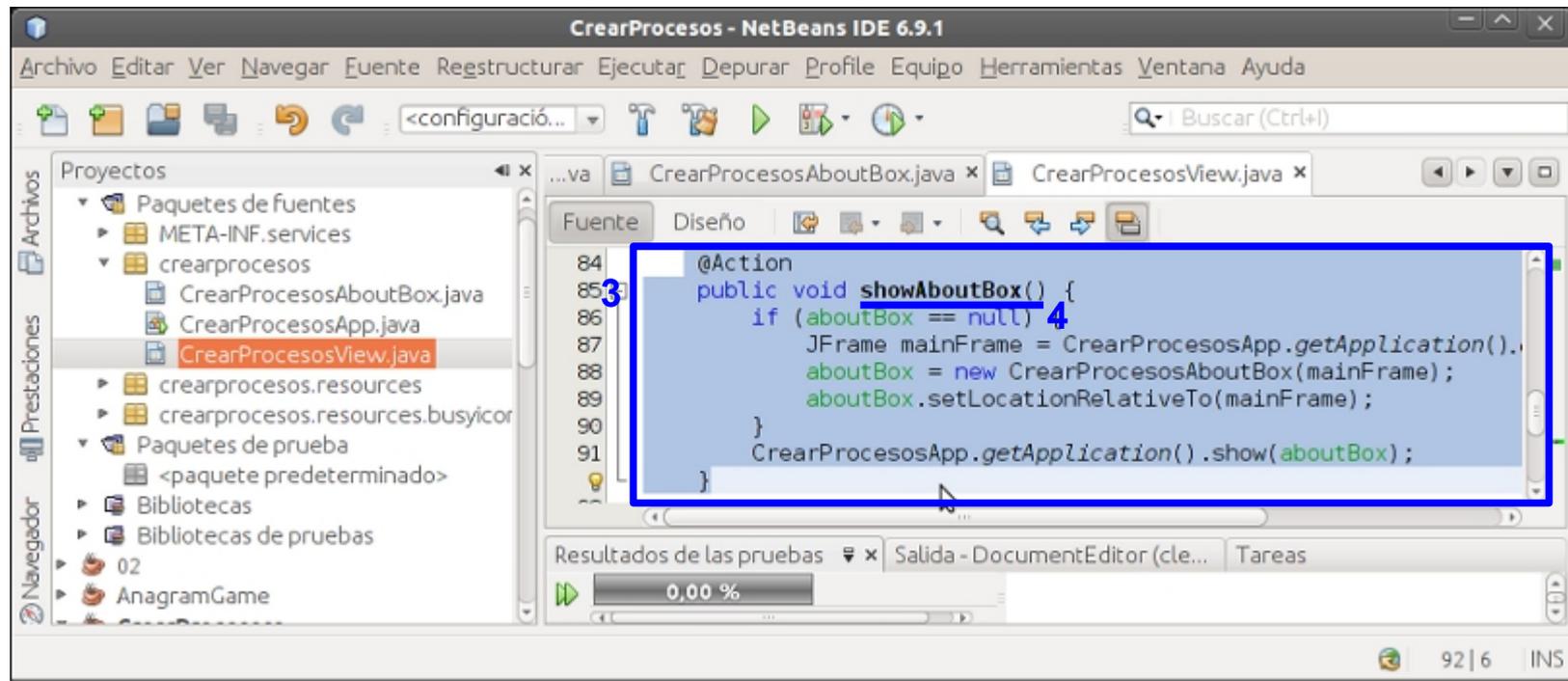
Vamos a incluir una acción (@Action), para poder asignarla como acción a realizar cuando el usuario haga clic sobre el botón:

2. En el código, buscamos la línea que comienza con @Action (en mi caso es la línea 84).

3. Copiamos esa acción y la pegamos debajo de ella.

Una acción es un método de la clase que tiene que ser público, no devuelve ningún valor y no recibe ningún parámetro.

4. Cambiamos el nombre de la acción showAboutBox() por otro, por ejemplo: crearNuevoEditor().



NetBeans Proyecto CrearProcesos. Incluyendo una nueva acción.

Solo nos queda modificar el código del método y asignar la acción al botón.

Implementar: Crear procesos.

Comprobamos que el comando con el que crearemos los procesos, funciona.

5. El comando que debemos ejecutar para lanzar el editor es:

```
java -jar C:\Users\usuario\Documents\NetBeansProjects\Editor\dist\DocumentEditor.jar
```

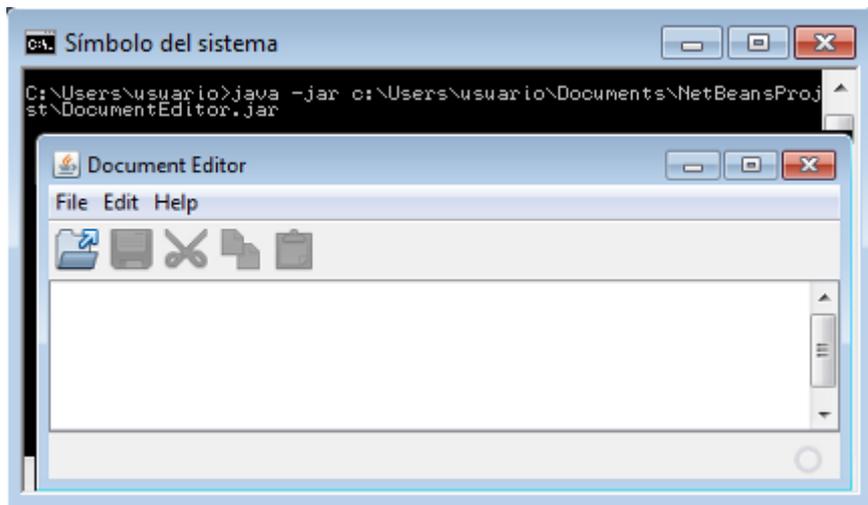
o

```
java -jar /home/usuario/NetBeansProjects/Editor/dist/DocumentEditor.jar
```

Podemos probarlo en un terminal de comandos y funcionará:

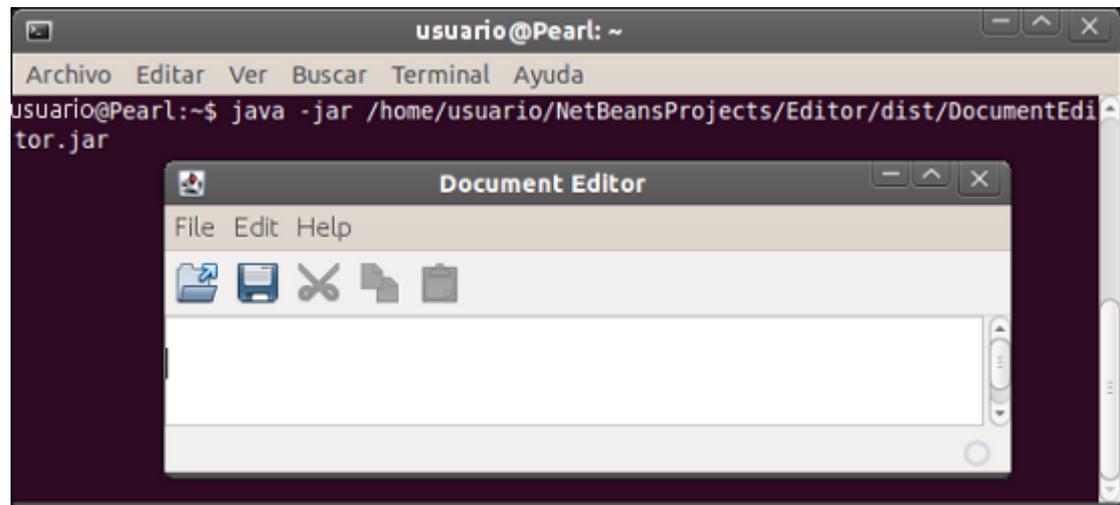
En Windows

Símbolo Windows → Todos los programas →
Accesorios → Símbolo del Sistema



En GNU/Linux

Aplicaciones → Accesorios → Terminal



Comenzamos a incluir el código necesario para crear procesos:

6. Necesitamos importar las clases `java.lang.Process` y `java.lang.Runtime`.

```
/* Clases necesarias para poder crear procesos */  
import java.lang.Process;  
import java.lang.Runtime;
```

Implementar: Crear procesos.

Implementamos el código de la acción:

7. No debemos olvidar:

- Definir una variable que referenciará al objeto de tipo Process. Esta variable nos permitirá comunicarnos con el proceso.
- El método `Runtime.getRuntime().exec(comando)`, nos devuelve un objeto de tipo Process y puede generar excepciones.
- Capturar las excepciones que se puedan producir.

8. Este es un ejemplo de código a implementar:

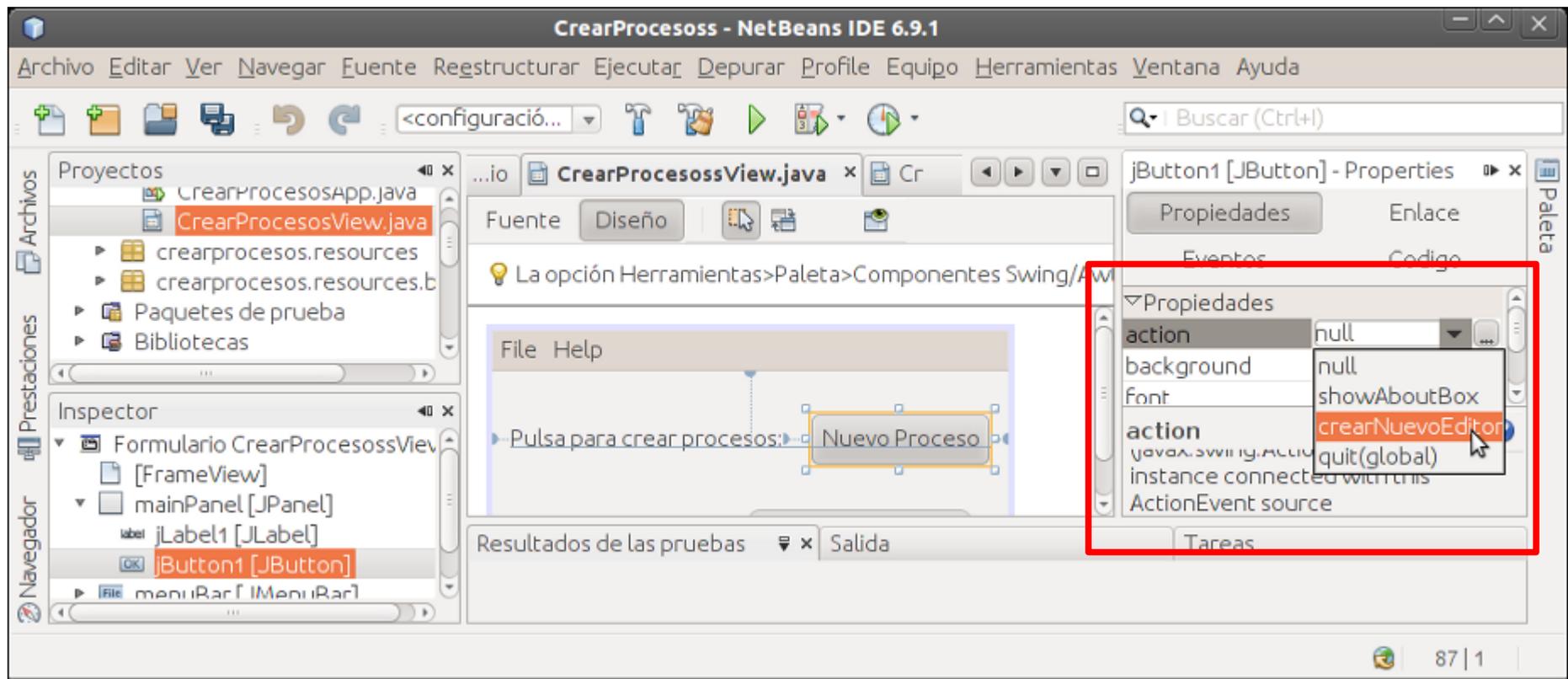
```
19  /* Clases necesarias para poder crear procesos */
20  import java.lang.Process;
21  import java.lang.Runtime;
22
23  /** ... */
24
25  public class CrearProcesosView extends FrameView {
26
27
28      public CrearProcesosView(SingleFrameApplication app) {...}
29
30
31      @Action
32      public void showAboutBox() {...}
33
34      @Action
35      public void crearNuevoEditor() {
36          Process nuevoProceso; //Definimos una variable de tipo Process
37          try{
38              nuevoProceso = Runtime.getRuntime().exec("java -jar "+
39                  "/home/usuario/NetBeansProjects/DocumentEditor/dist/DocumentEditor.jar")
40              //Lanzamos la ejecución del .jar del editor de textos
41          }catch (SecurityException ex){
42              System.out.println("Ha ocurrido el siguiente error de seguridad. " +
43                  "No se ha podido crear el proceso por falta de permiso.");
44          }catch (Exception ex){
45              System.out.println("Ha ocurrido un error, descripción: "
46                  + ex.toString());
47          }
48      }
49  }
```

Implementar: Crear procesos.

Asignamos la acción al **botón**:

7. Volvemos al Diseño de nuestra ventana de aplicación o formulario.

8. Seleccionamos el botón y en sus propiedad **action** establecemos el valor crearNuevoEditor. Esto hará que cada vez que se hace clic sobre el botón se ejecutarán las instrucciones que hemos implementado.

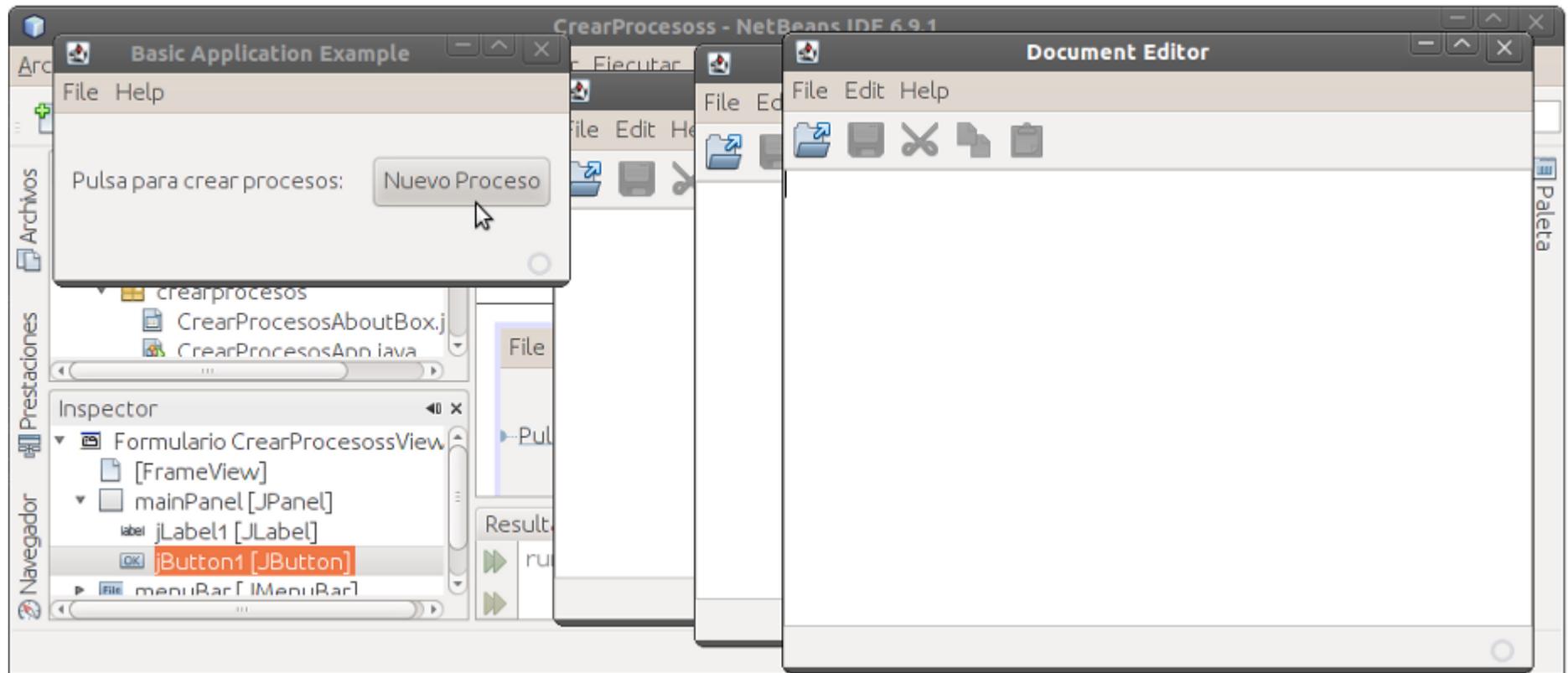


NetBeans Proyecto CrearProcesos. Asignando la acción crearNuevoEditor como action del botón.

Probar el proyecto

Para finalizar probamos su funcionamiento:

9. Menu Ejecutar → Ejecutar Main Project (F6).



NetBeans Proyecto en ejecución CrearProcesos y 3 procesos de tipo DocumentEditor creados por él.

Posibles mejoras

Como cualquier otro, este código es muy mejorable. Aquí incluimos algunas de las posibles mejoras:

1. Crear código que sea independiente de la plataforma.
2. Crear una estructura, por ejemplo una lista, que nos permita tener referenciados todos los objetos proceso que vayamos creando. En nuestro ejemplo estamos utilizando siempre la misma variable (`nuevoProceso`); y además, está definida dentro del método `crearNuevoEditor`; lo que no nos permite comunicarnos con ese proceso desde el resto del código del proyecto; y aún más, sólo nos permitiría comunicarnos con el último creado.
3. Poder crear cualquier tipo de proceso, permitiéndole al usuario introducir el comando de ejecución del proceso.

Recuerda, esto implicará incluir, por ejemplo, un cuadro de texto, y chequear que ese comando esté bien construido.

Nosotros vamos a mostrar a continuación una posible implementación para la primera de las mejoras propuestas.

Posibles mejoras

Posible implementación de la primera mejora: “Crear código que sea independiente de la plataforma”.

Puedes ver en la captura, como:

1. Obtenemos el nombre del sistema operativo en el que se está ejecutando la aplicación.
2. Dependiendo del sistema operativo, lanzamos un comando u otro.

La diferencia entre uno y otro, está en la ruta de acceso al .jar del Editor (y las barras de directorio, en Windows es “\” y en GNU/Linux es “/”).

En el caso de Windows incluimos \\ para evitar que sea interpretado como carácter de escape.

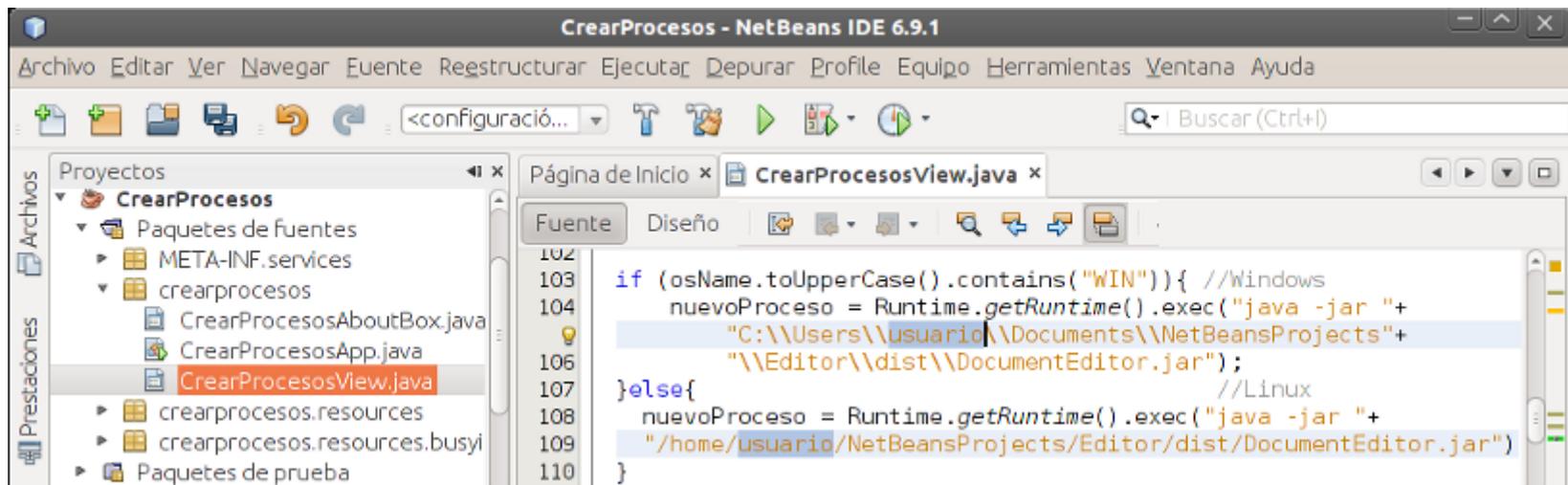
Importante es resaltar, que este ejemplo funciona sólo para el usuario “usuario”.

```
19  /* Clases necesarias para poder crear procesos */
20  import java.lang.Process;
21  import java.lang.Runtime;
22
23  /** ... */
24
25  public class CrearProcesosView extends FrameView {
26
27      public CrearProcesosView(SingleFrameApplication app) {...}
28
29      @Action
30      public void showAboutBox() {...}
31
32      @Action
33      public void crearNuevoEditor() {
34          Process nuevoProceso; //Definimos una variable de tipo Process
35          try{
36              // Obtememos el nombre del SO
37              String osName = System.getProperty("os.name");
38
39              if(osName.toUpperCase().contains("WIN")){ //Windows
40                  //Lanzamos la ejecución del .jar del editor de textos
41                  nuevoProceso = Runtime.getRuntime().exec("java -jar "+
42                      "C:\\Users\\usuario\\Documents\\NetBeansProjects"+
43                      "\\Editor\\dist\\DocumentEditor.jar");
44              } else { //Linux
45                  //Lanzamos la ejecución del .jar del editor de textos
46                  nuevoProceso = Runtime.getRuntime().exec("java -jar "+
47                      "/home/usuario/NetBeansProjects/Editor/dist/DocumentEditor.jar");
48              }
49          } catch (SecurityException ex){
50              System.out.println("Ha ocurrido el siguiente error de seguridad. " +
51                  "No se ha podido crear el proceso por falta de permiso.");
52          } catch (Exception ex){
53              System.out.println("Ha ocurrido un error, descripción: "
54                  + ex.toString());
55          }
56      }
57  }
```

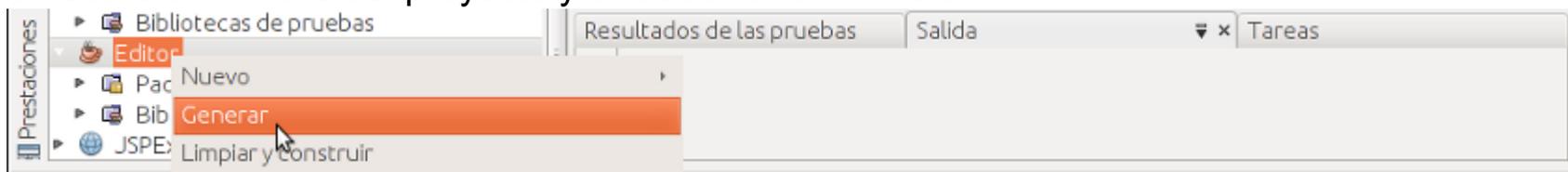
Recomendaciones para probar el ejemplo

Si no has ido construyendo paso a paso el ejemplo, puedes probar el ejemplo que te suministramos implementado en la plataforma. Pero debes tener en cuenta que el código no está preparado para que lo ejecute un usuario con cualquier nombre de usuario. Deberás modificar un poco el código del proyecto CrearProcesos antes de lanzar su ejecución.

1. **Cambia en la ruta de acceso al .jar, el nombre del “usuario”, por tu nombre de usuario en el sistema.**



2. Si no está generado el ejecutable del Editor, puedes generarlo haciendo clic con el botón secundario del ratón sobre el nombre del proyecto y seleccionando Generar.



Credenciales

Imagen	Datos de licencia
<p>Todas las capturas de pantalla de esta presentación, tienen como datos de licencia:</p> <p>Autoría: Margarita I. Nieto Castillejo Licencia: Uso educativo-no comercial. Procedencia: Capturas de pantalla del IDE NetBeans 6.9.1, terminales de comandos de los sistemas operativos Windows 7 y Ubuntu 10.10.</p>	