

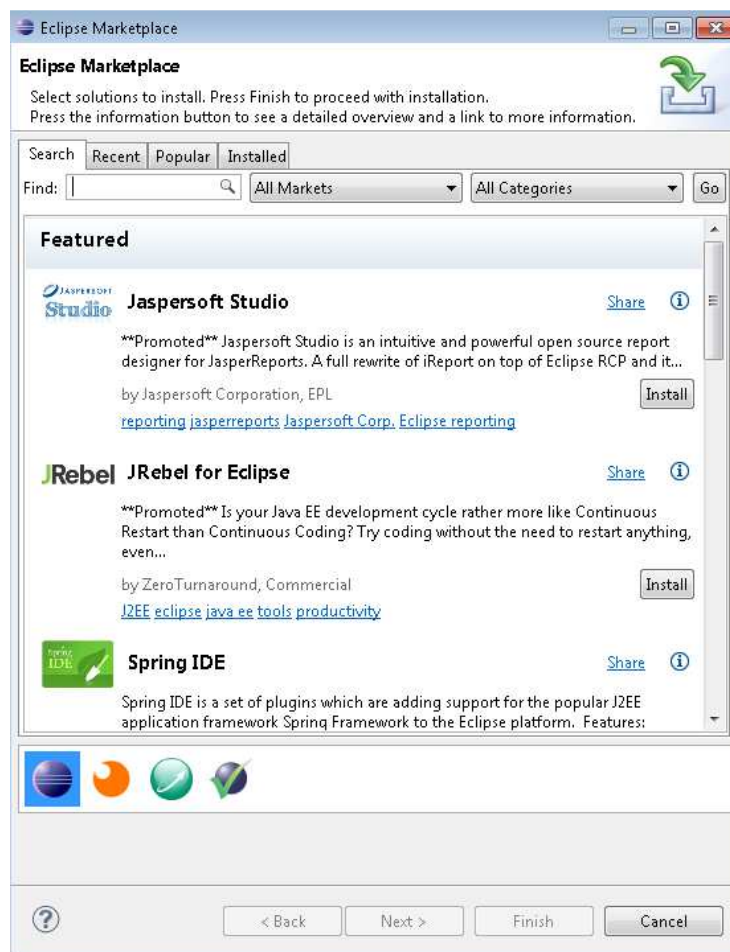
INSTALAR PLUGINS EN ECLIPSE

Eclipse es uno de los entornos de desarrollo integrados más populares (IDE) en la actualidad. El IDE compatible con una amplia variedad de plugins, que puede ser añadido a la misma. A través de ellas, los desarrolladores pueden ampliar las capacidades de Eclipse y personalizar el IDE para satisfacer sus necesidades particulares. Instalar y desinstalar plugins en Eclipse, nos permitirá aumentar sus funcionalidades. Para instalar **plugins** en **eclipse** hay varias formas, dependiendo del plugin.

1. INSTALACION DE PLUGINS CON MARKETPLACE

Para instalar plugins con Marketplace haremos lo siguiente:

- Abrimos Eclipse.
- Pinchamos en **Help -> Eclipse Marketplace**. Aquí podremos buscar plugins por categorías o por su nombre.



- Pinchamos en **Install**.

- Nos mostrara lo que contiene el plugin.
- Nos pide que aceptemos los términos de licencia.
- Comenzara la instalación.
- Cuando termine nos pedirá que reiniciemos el programa.
- El plugin ya estará instalado y se configurara según el plugin.

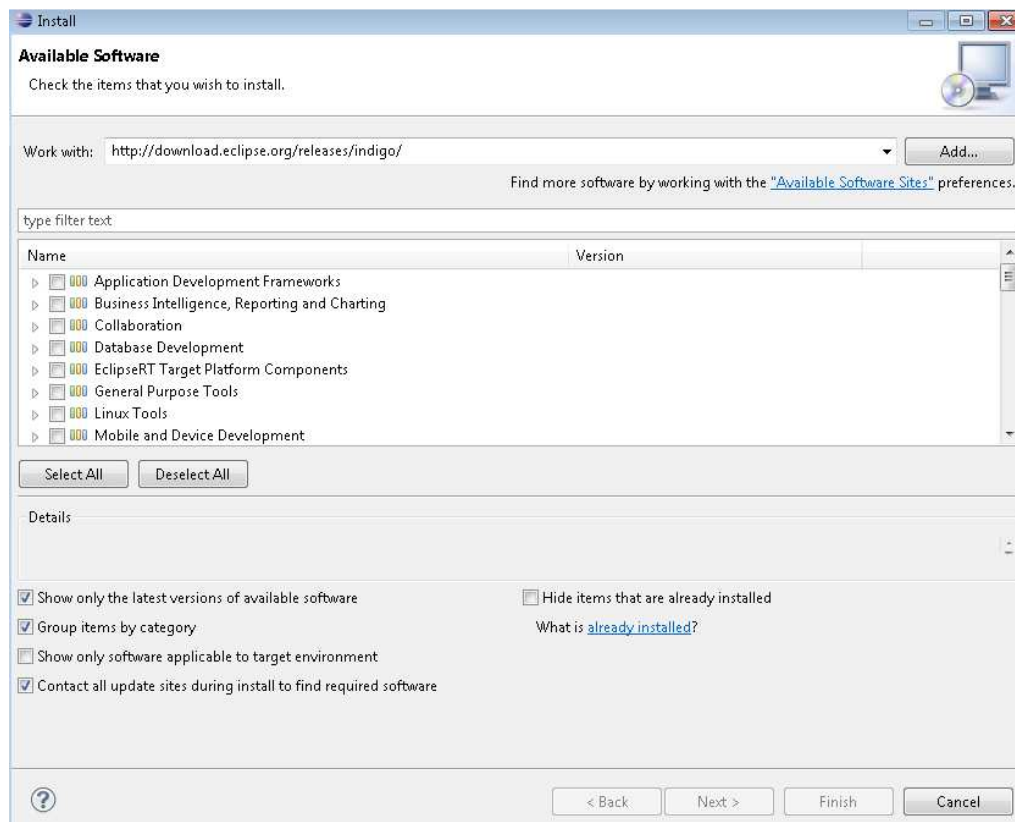
Para desinstalar plugins con MarketPlace haremos lo siguiente:

- Abrimos Eclipse.
- Pinchamos en **Help -> Eclipse Marketplace -> Installed.**
- Buscamos el plugin que queremos desinstalar.
- Pinchamos en Uninstall.
- Aceptamos que queremos desinstalarlo.
- Cuando termine nos pedirá que reiniciemos el programa.
- El plugin ya estará desinstalado.

2. INSTALACIÓN DE PLUGINS DESDE UN REPOSITORIO

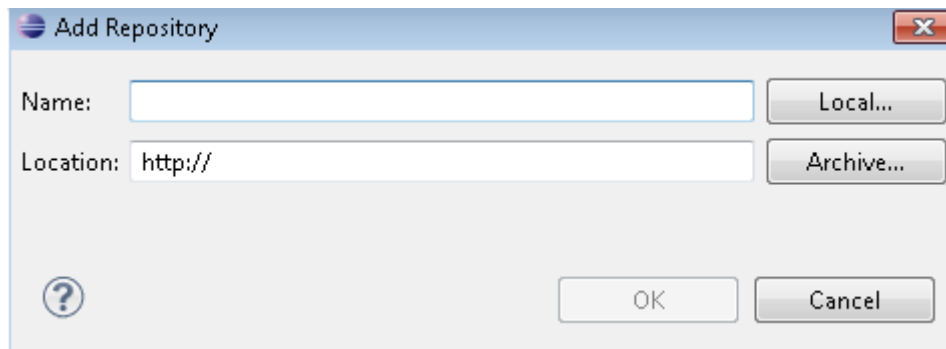
Para instalar plugins desde un repositorio, sabiendo su dirección, haremos lo siguiente:

- Abrimos Eclipse.
- Pinchamos en **Help -> Install New Software**



- En **Work with** copiamos la dirección del repositorio y en el cuadro de texto podemos filtrar el contenido del repositorio. Si ese repositorio lo vamos a usar muy a menudo, lo podemos guardar con un nombre. Para ello, pinchamos en

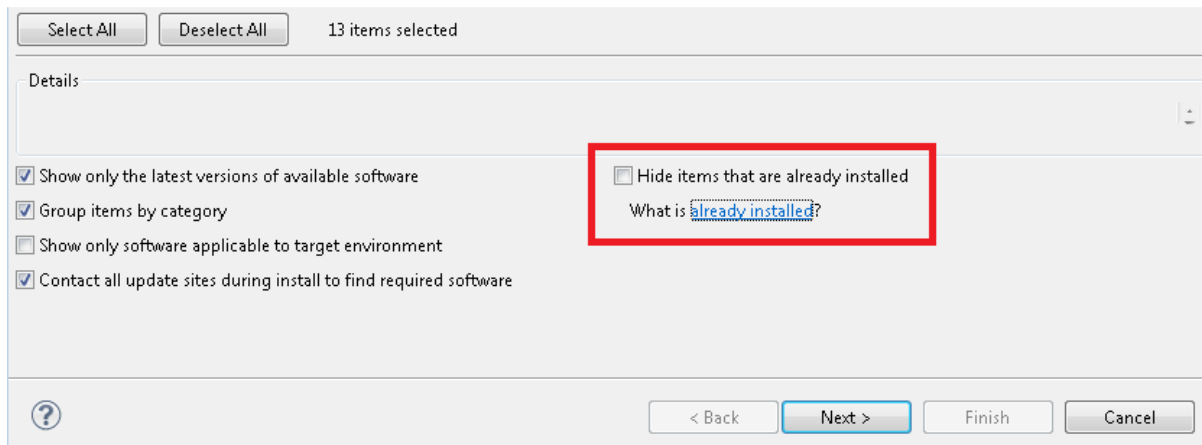
Add y nos aparecerá un dialogo donde debemos indicar el nombre (opcional) y su localización (la dirección del repositorio). Estos aparecerán en la lista de **Work with**.



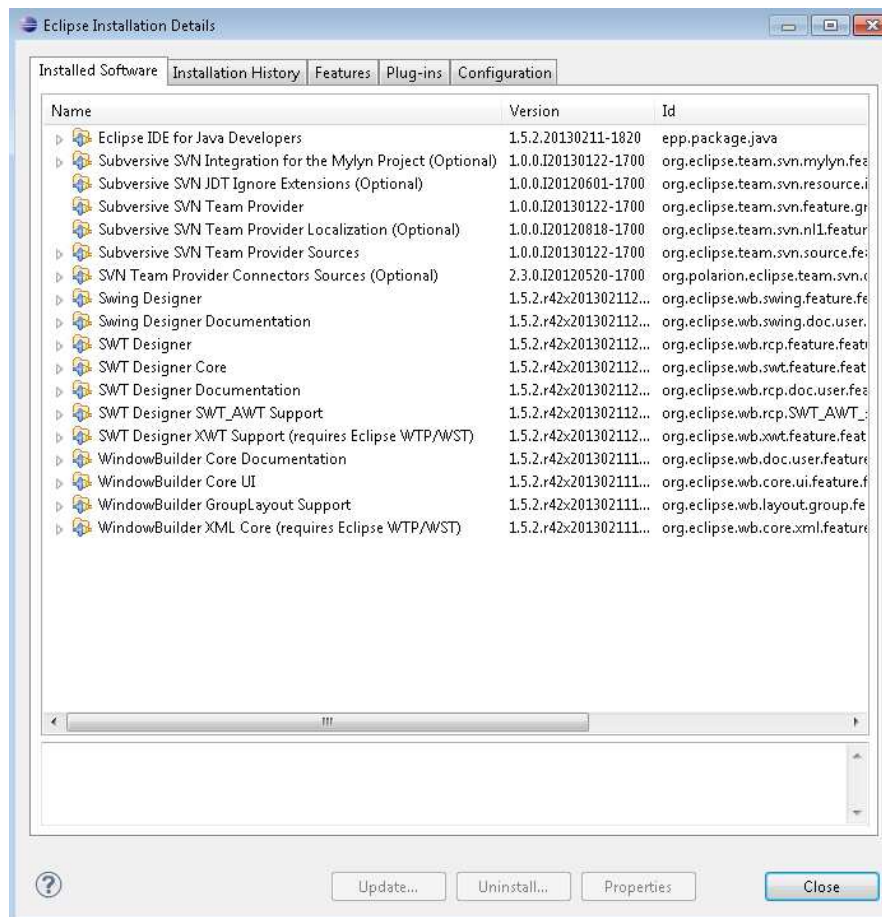
- Cuando encontremos los plugins necesarios, los seleccionamos y pinchamos en **Next**.
- Nos mostrara lo que contiene el plugin.
- Nos pide que aceptemos los términos de licencia.
- Comenzara la instalación.
- Cuando termine nos pedirá que reiniciemos el programa.
- El plugin ya estará instalado y se configurara según el plugin.

Para desinstalar plugins instalados desde un repositorio, haremos lo siguiente:

- Abrimos Eclipse.
- Pinchamos en **Help -> Install New Software**
- Pinchamos en el enlace “**already installed**”.



- Veremos la lista de los plugins instalados, seleccionamos los que queremos desinstalar.



- Pinchamos en Uninstall
- Aceptamos que queremos desinstalarlo.
- Cuando termine nos pedirá que reiniciemos el programa.
- El plugin ya estará desinstalado.

NOTA: Esta opción de instalación se puede realizar directamente seleccionando el archivo ZIP del plugin directamente desde una ubicación local, con la opción ARCHIVO.

3.- PLUGINS CON INSTALACIÓN MANUAL

Para instalar plugins manualmente haremos lo siguiente:

- Descargamos un plugin y lo descomprimos.
- Abrimos la carpeta **plugins** buscamos el plugin que necesitamos (puede ser mas de uno).
- Los copiamos a la carpeta **plugins** dentro de nuestro eclipse.
- Hacemos lo mismo en la carpeta **feature**.
- Reiniciamos el programa.

NOTA: no se llaman igual los de la carpeta **plugins** y **feature** pero si tienen el mismo código en su nombre

Para desinstalar plugins manualmente haremos lo siguiente:

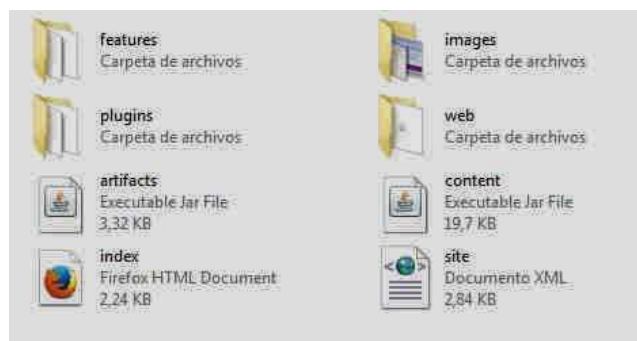
- Borramos los archivos copiados anteriormente, tanto en **plugins** como en **feature**.
- Reiniciamos el programa.

3. PLUGIN WINDOWBUILDER

Es un Plugin de Eclipse que nos permite desarrollar rápidamente una interfaz Gráfica de Usuario (GUI) por medio de una paleta de componentes con tan solo arrastrar y soltar, generándonos el código necesario para su funcionamiento...

Para instalarlo se debe obtener de la página oficial (<http://www.eclipse.org/windowbuilder/download.php>) el enlace o como en este caso el empaquetado del plugin. Es importante saber que debemos escoger el enlace o zip correspondiente a nuestra versión de Eclipse...

Cuando descarguemos el archivo zip, lo descomprimos y vemos el contenido, de aquí debemos sacar las librerías necesarias para vincular al Eclipse...



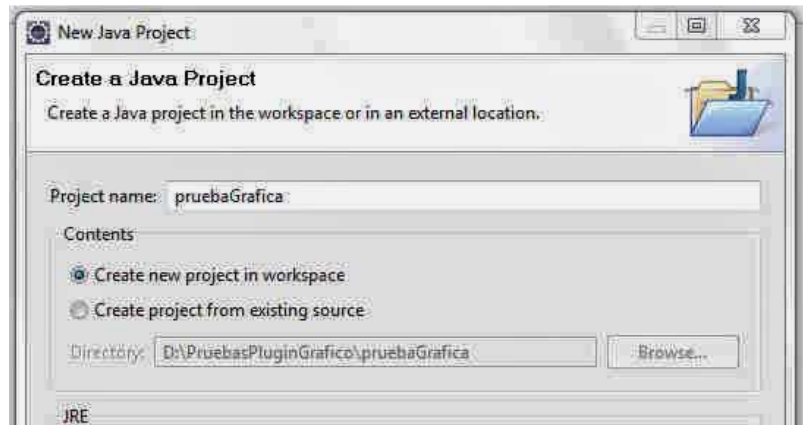
Copiamos el contenido de las carpetas features y plugin en las carpetas de instalación de Eclipse con el mismo nombre....



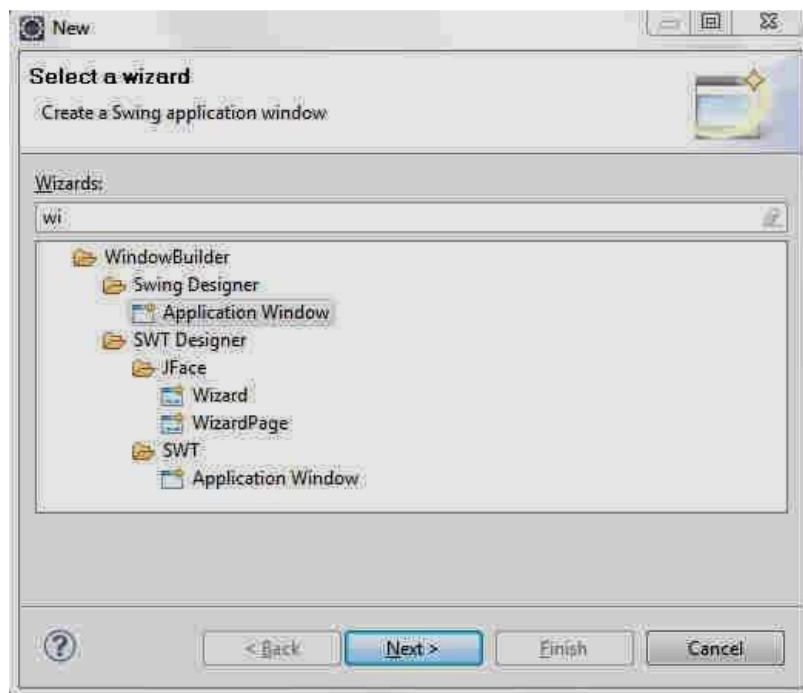
Y listo, de esta manera con solo copiar las librerías ya finalizamos el proceso de instalación, ahora solo es cuestión de iniciar Eclipse y crear nuestra primera aplicación por medio del Editor...

CREAR UNA VENTANA

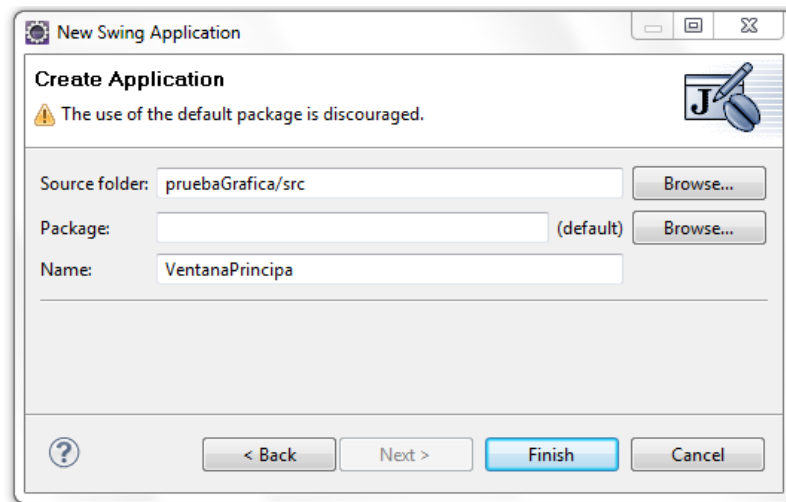
El uso de la herramienta es demasiado simple, para verificarlo crearemos un Java Project normal (File/New/Java Project...)



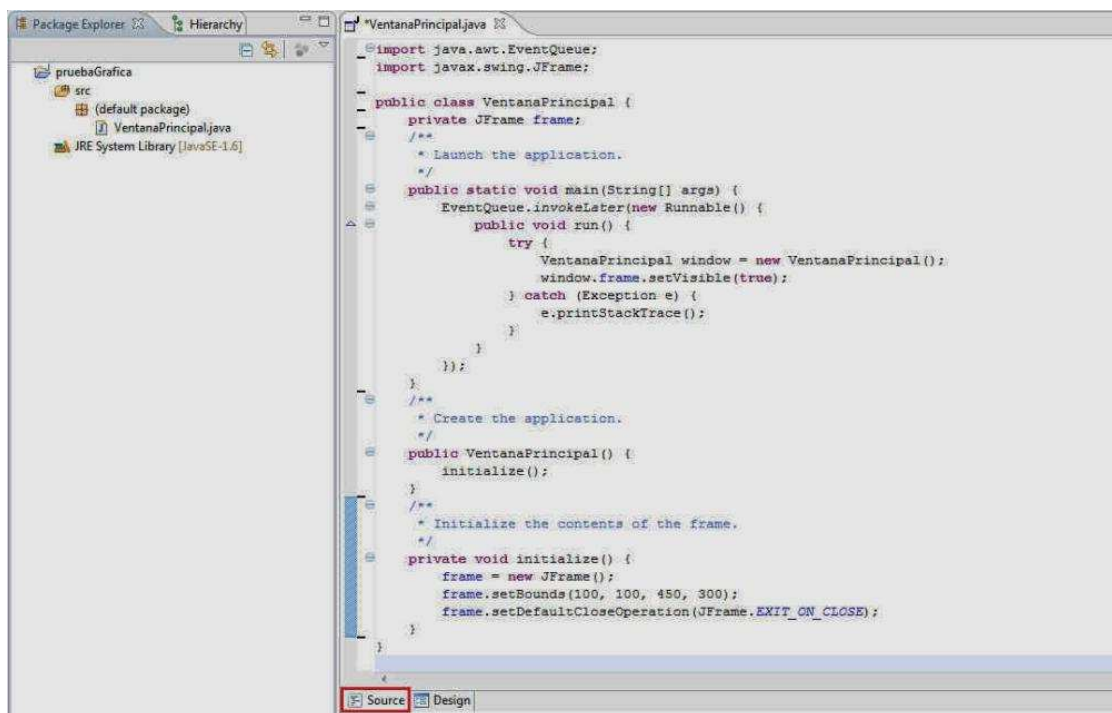
Ahora crearemos nuestra ventana, para eso damos clic derecho en el proyecto o vamos a File/new/Other... y buscamos WindowBuilder, posteriormente ingresamos a Swing Designer y seleccionamos Application Window y damos next...



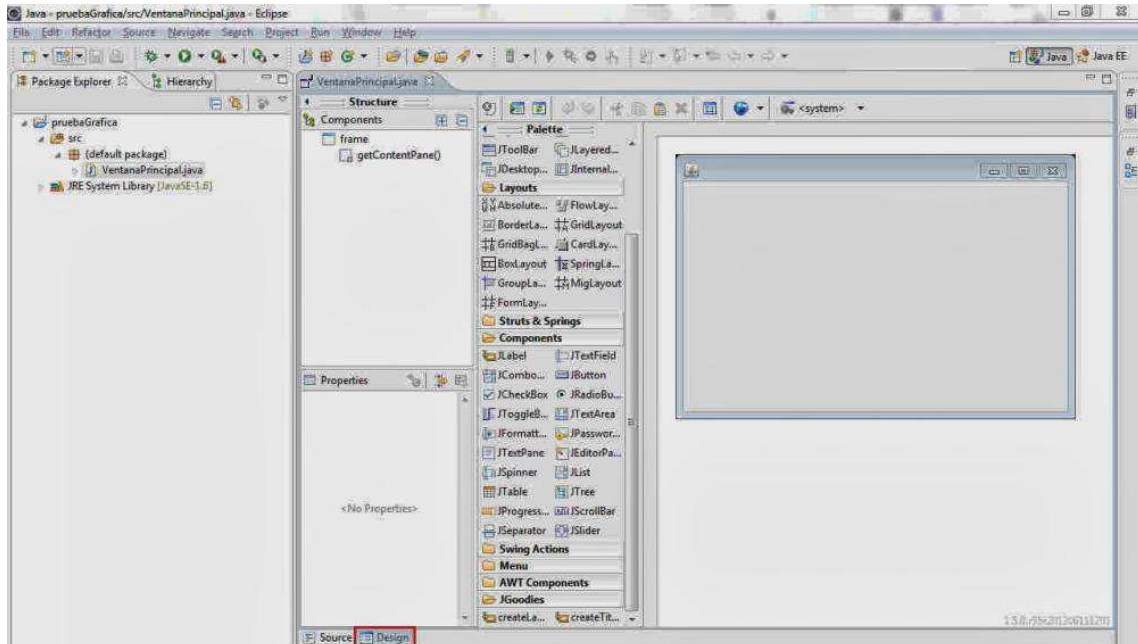
Al hacer esto se carga otra ventana donde está definido el proyecto... ingresamos el nombre de nuestra ventana y le damos finish...



Automáticamente se crea una clase con el nombre que definimos y el código necesario para nuestra ventana...

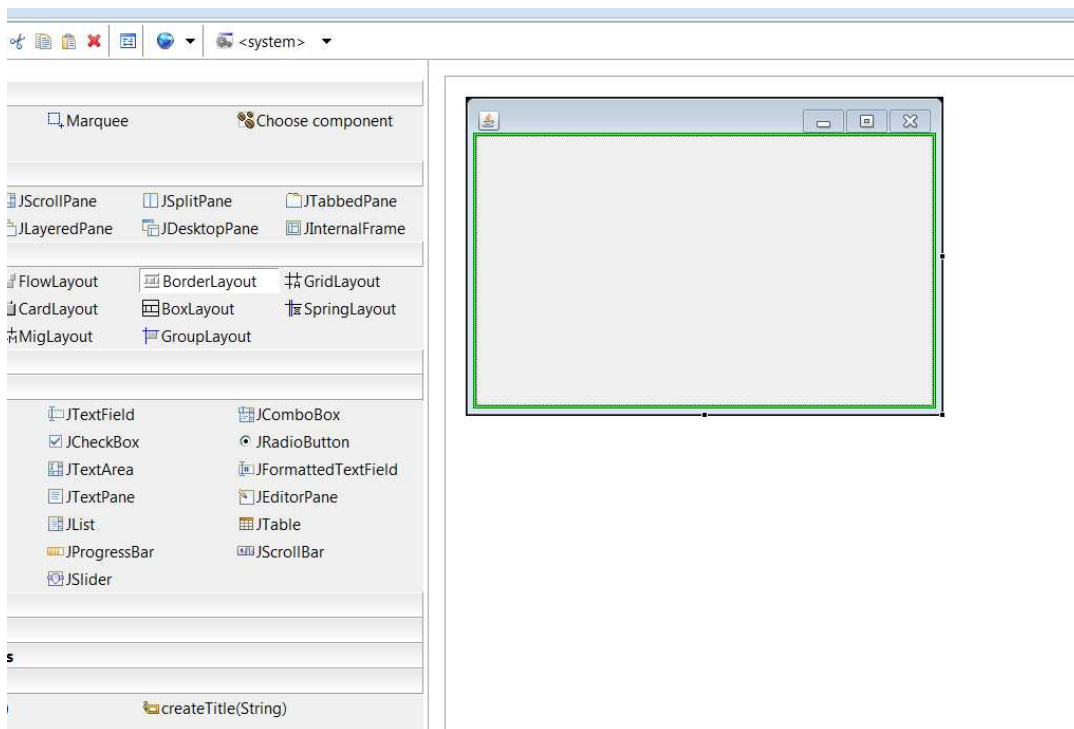


También podemos ver 2 pestañas, source donde nos encontramos y vemos el código y la otra Design que nos permitirá ver la paleta de componentes para crear nuestras aplicaciones de forma gráfica...

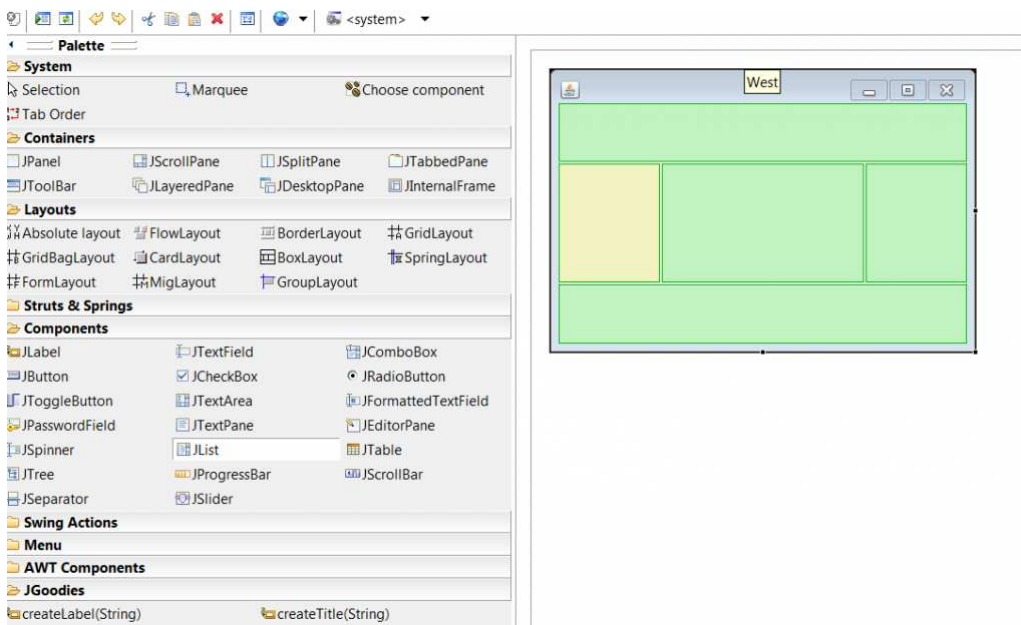


Encontramos la paleta de componentes, el árbol de componentes usados, las propiedades y la parte gráfica donde vemos la ventana que estamos diseñando, debemos saber que estos paneles pueden ser cambiados de posición como cualquier panel de los usados en Eclipse, además podemos modificar el código y nuestra parte gráfica se actualiza según la modificación...

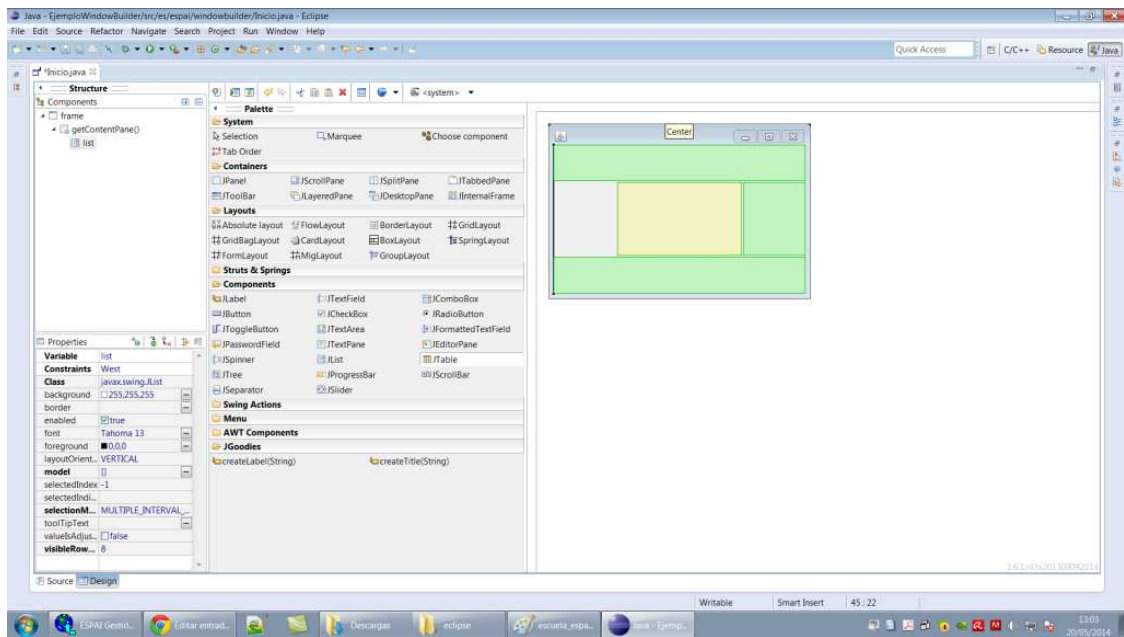
Esta es la ventana principal de WindowBuilder. Como veis, nos permite añadir elementos gráficos como imágenes, botones o listas a nuestro programa, a parte de otros muchos.



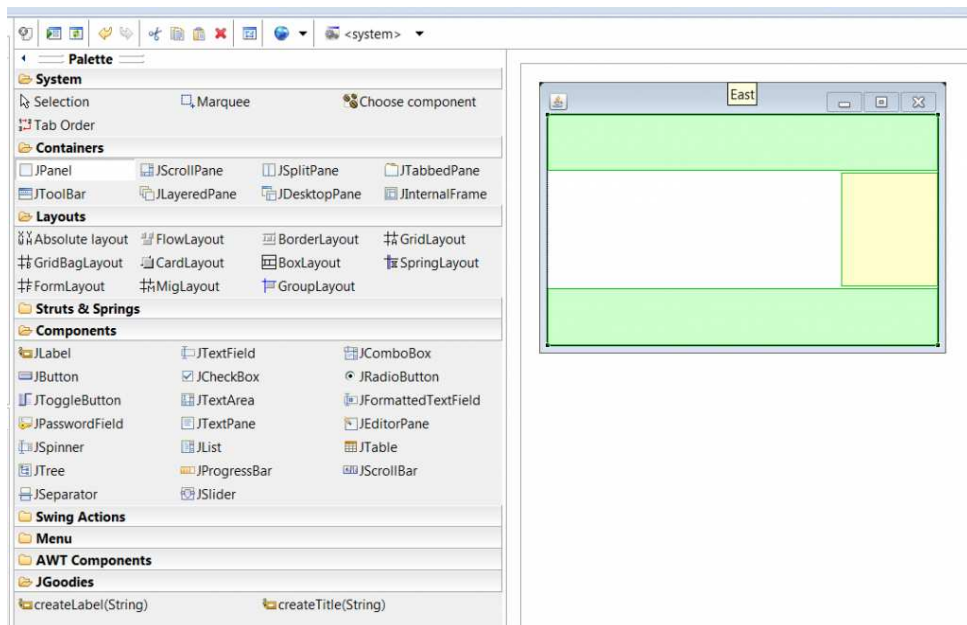
Marcamos el elemento BorderLayout y hacemos click en nuestra ventana. Esto lo que hace es asignar una forma específica de distribuir los elementos en la pantalla, en concreto en 5 secciones: Norte, sur, este, oeste y centro.



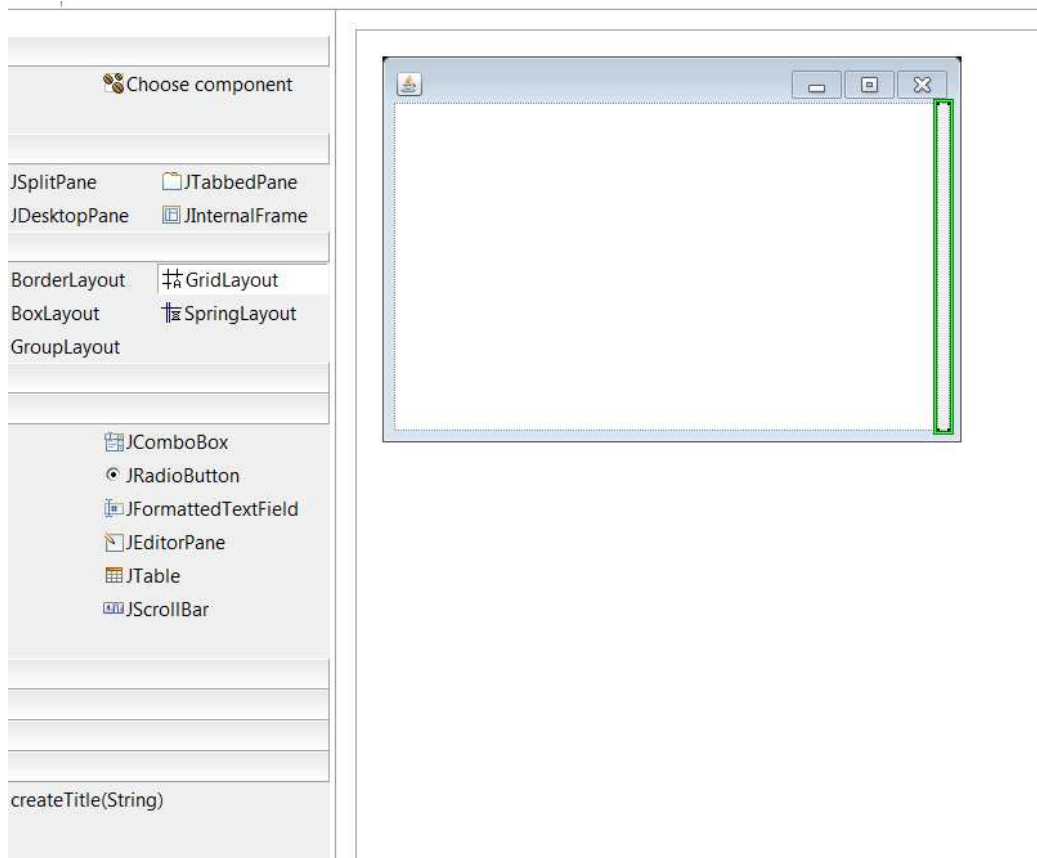
Añadimos un JList (lista) a la izquierda. Como en el caso anterior, click en JList y luego click en la zona de la pantalla donde lo queréis (marcado en amarillo, aparecerá cuando pongáis el cursor encima)



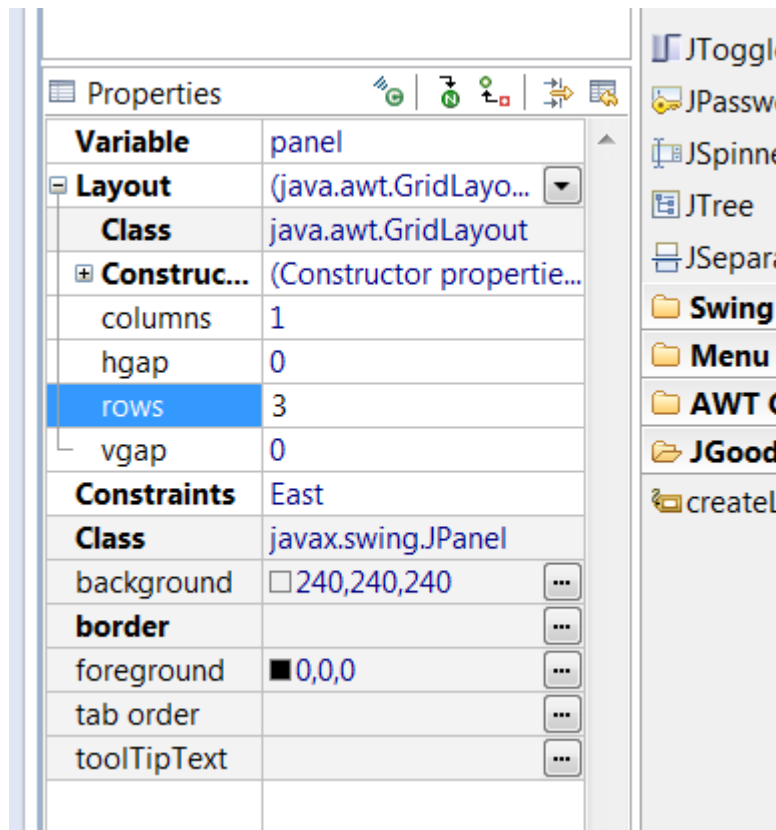
Añadimos un JTable en el centro. Un JList tiene una sola dimensión (una columna y varias filas) mientras un JTable tiene varias filas y columnas, similar a una hoja de cálculo.



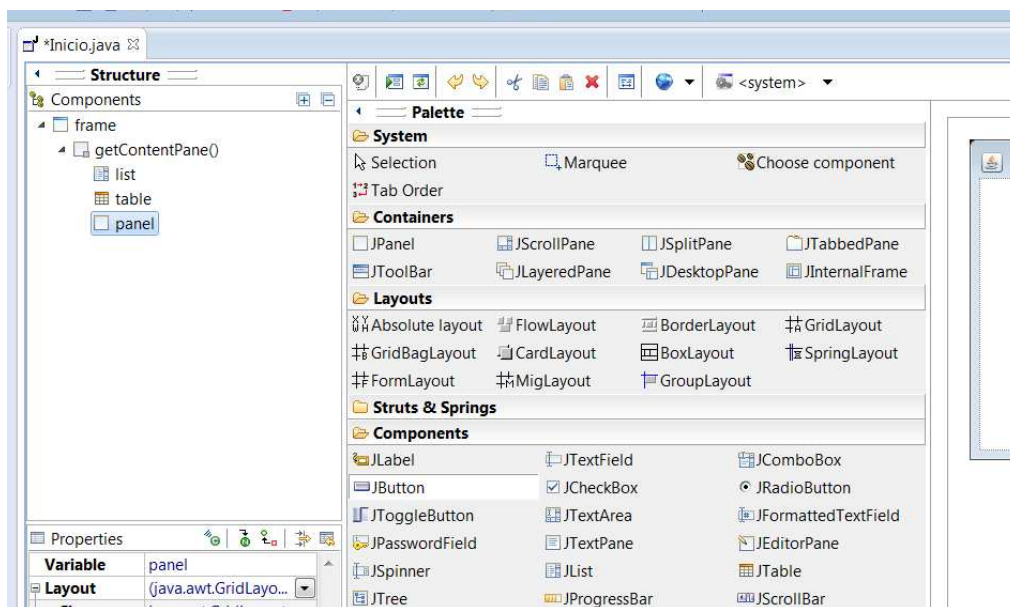
Añadimos un JPanel a la derecha. Mismo proceso que con el JList.



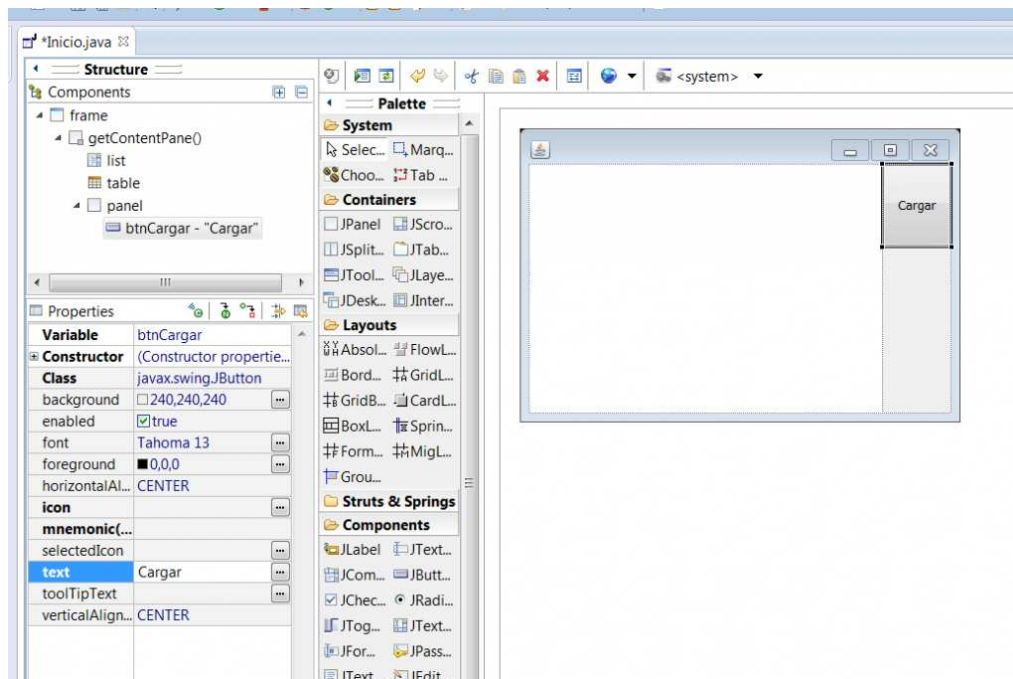
Cambiamos las propiedades de ese panel. hacemos que tenga un layout (distribución) tipo GridLayout



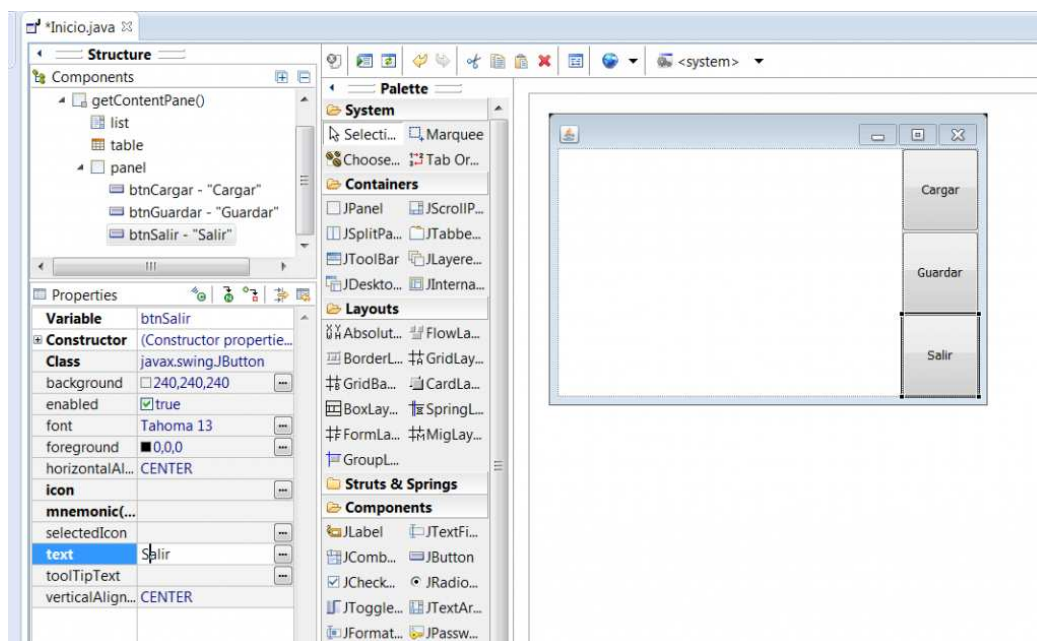
Vamos a las propiedades del layout, y lo hacemos de 1 columna y 3 filas.



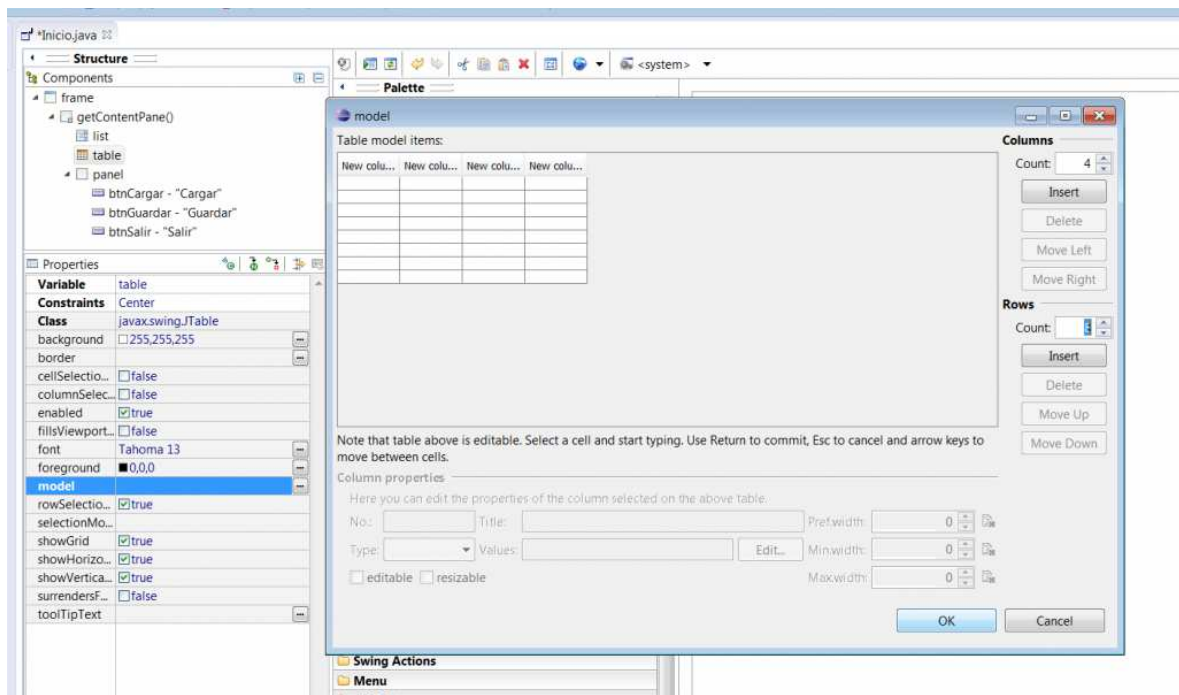
Añadimos un botón al JPanel. Si veis que no se ilumina el panel cuando os colocáis sobre la ventana, dejad el botón sobre el elemento “panel” que encontrareis a la izquierda, debajo de “table”. haced click sobre JButton, y luego sobre panel, y se añadirá el botón al panel.



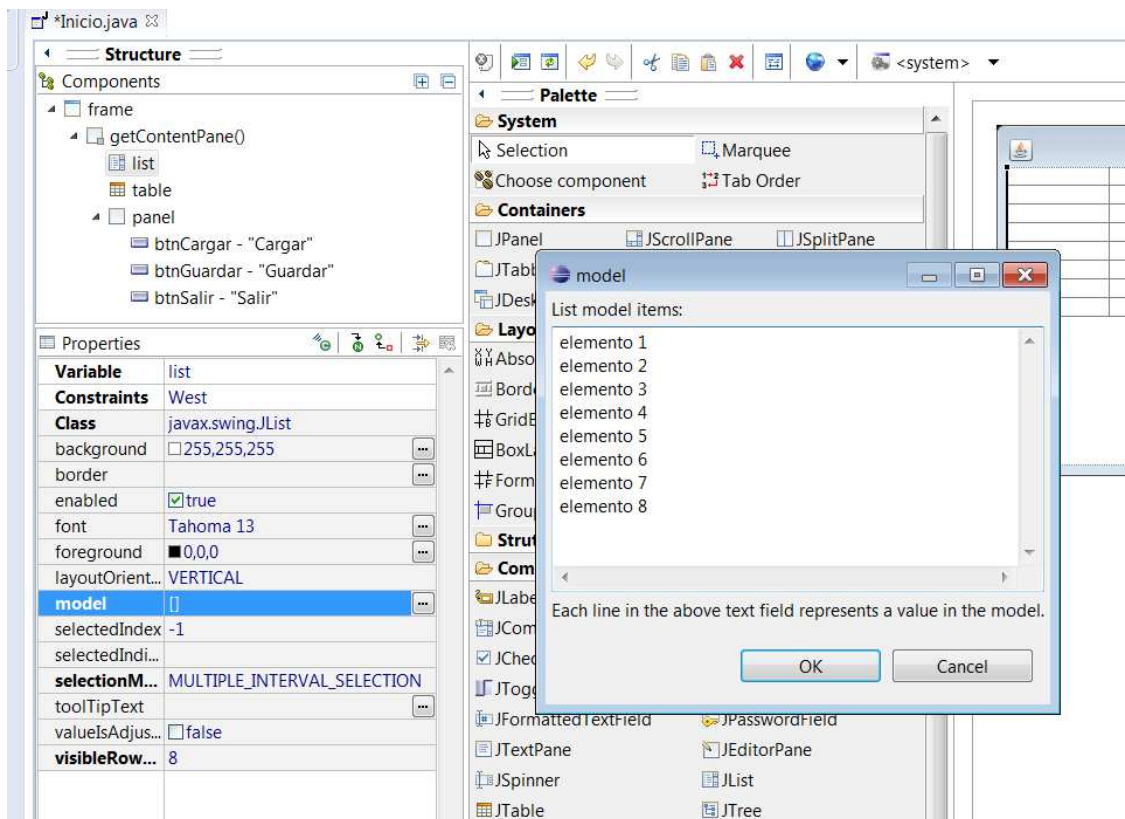
Cambiamos las características de ese botón. Le damos como nombre “btnCargar”, y como texto “Cargar”. El nombre será la variable que encontraremos luego en el código, y el texto es la descripción que aparecerá sobre el botón. Lo alineamos con “Center”.



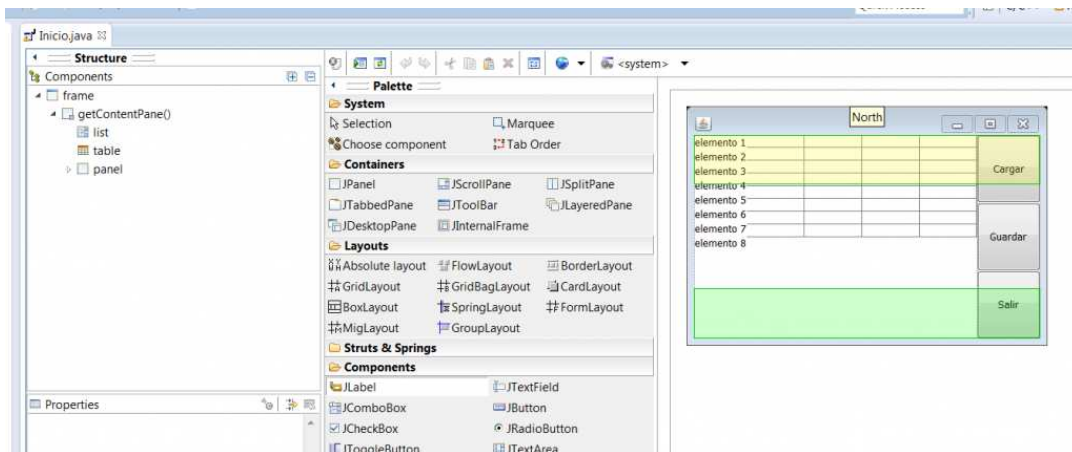
Repetimos el mismo proceso con los botones “Guardar” y “Salir”



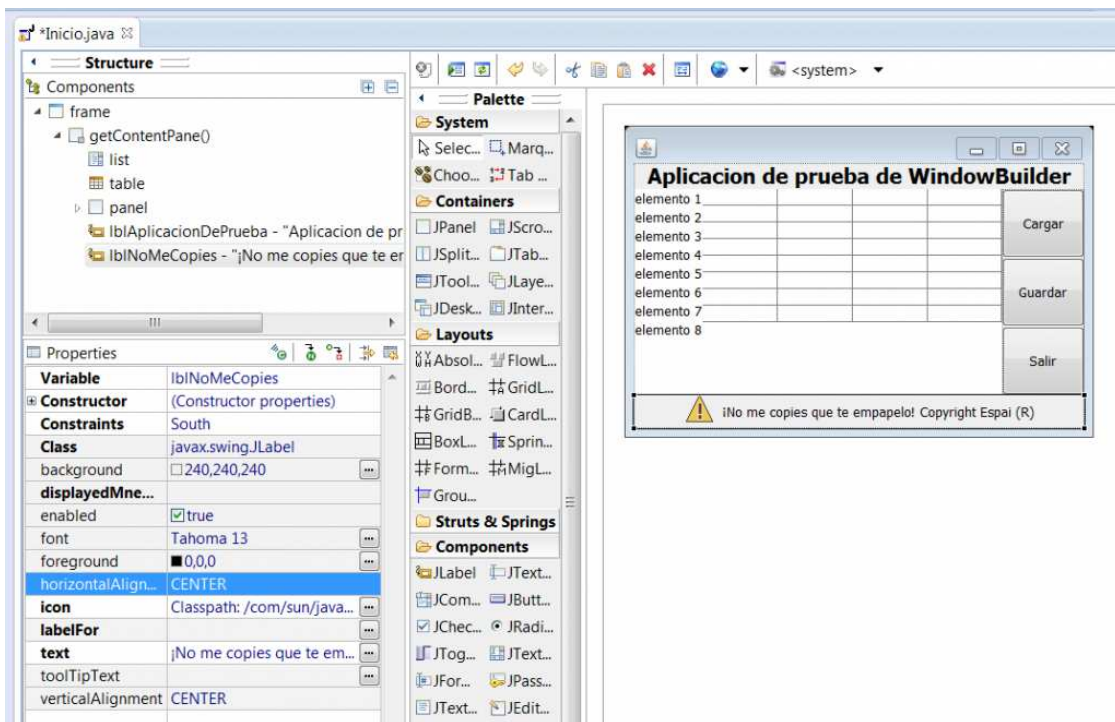
En las propiedades del JTable, pulsamos en los “...” a la derecha de “model”. Sale una ventana que nos permite añadir filas y columnas. Añadimos 8 filas y 4 columnas, o las que queráis.



hacemos lo mismo con el JList.



Hacemos click sobre JLabel, es decir añadimos una etiqueta en el “norte” de nuestra aplicacion, la parte superior. El texto será “Aplicacion de prueba de WindowBuilder”



hacemos lo mismo con el inferior. Además, le añadimos un icono haciendo click en “icon”.

```

*Inicio.java
    {null, null, null, null},
    {null, null, null, null},
    {null, null, null, null},
    {null, null, null, null},
    {null, null, null, null},
    {null, null, null, null},
    {null, null, null, null},
    {null, null, null, null},
    },
    new String[] {
        "New column", "New column", "New column", "New column"
    }
));
frame.getContentPane().add(table, BorderLayout.CENTER);

JPanel panel = new JPanel();
frame.getContentPane().add(panel, BorderLayout.EAST);
panel.setLayout(new GridLayout(3, 1, 0, 0));

btnCargar = new JButton("Cargar");
panel.add(btnCargar);

btnGuardar = new JButton("Guardar");
panel.add(btnGuardar);

btnSalir = new JButton("Salir");
panel.add(btnSalir);

lblAplicacionDePrueba = new JLabel("Aplicacion de prueba de WindowBuilder");
lblAplicacionDePrueba.setFont(new Font("Tahoma", Font.BOLD, 20));
lblAplicacionDePrueba.setHorizontalAlignment(SwingConstants.CENTER);
frame.getContentPane().add(lblAplicacionDePrueba, BorderLayout.NORTH);

lblNoMeCopies = new JLabel("\u00A1No me copies que te empapelo! Copyright Espai (R)");
lblNoMeCopies.setHorizontalAlignment(SwingConstants.CENTER);
lblNoMeCopies.setIcon(new ImageIcon(Inicio.class.getResource("/com/sun/java/swing/plaf/windows/ic
frame.getContentPane().add(lblNoMeCopies, BorderLayout.SOUTH);
}
}

```

haced click en la pestaña “design” para ver el código.

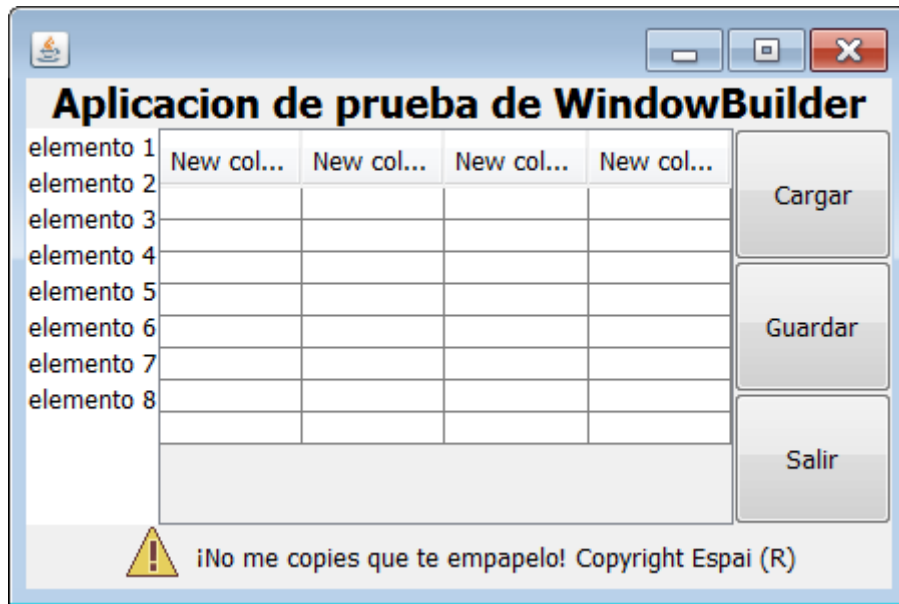
```

    }
));
frame.getContentPane().add(new JScrollPane(table), BorderLayout.CENTER);

```

Esta parte hay que añadirla a mano. Si ejecutais la aplicación ahora (haced la prueba) veréis que no salen los títulos de las columnas, y la tabla aparece cortada. Por ello le añadimos el fragmento de código que aparece en la captura. Lo que hacemos es “envolver” el JTable con un JScrollPane, que es un elemento gráfico que añade precisamente eso: barras de desplazamiento y encabezados a las tablas.

Para hacer esto, debéis ir a la pestaña “Source” donde podréis realizar este cambio.



Como veis, al ejecutar la aplicación ya aparece la tabla correctamente.

4.- OTROS EJEMPLOS DE PLUGINS PARA ECLIPSE

Algunos de los plugins más utilizados en el entorno Eclipse son:

[Subclipse](#) y [Subversive](#): Ambos plugins manejan los repositorios de subversión. Además, también se encargan de las tareas más importantes relacionadas con el control de versiones.

[EGit](#): Éste utiliza el sistema de control de versiones Git y es un plugin muy popular para la gestión de los códigos y los proyectos de código de los usuarios. También utiliza la biblioteca JGit, que es un núcleo basado en Java que se puede descargar por separado.

[m2eclipse](#): ¿Has oído hablar de la herramienta de construcción Apache Maven? Este gestor de proyectos y herramienta de comprensión puede ser integrado en Eclipse con el plugin de m2eclipse.

[Eclipse Marketplace Client](#): Este cliente ya esté instalado en su sistema. La Fundación Eclipse, la compañía detrás del IDE, ha estado haciendo su Market de una parte del propio software. Realiza un seguimiento de las descargas y realizar otras actividades.

[FindBugs](#): Se trata de un auditor que le da una ventaja extra para encontrar errores. El plugin viene con numerosos patrones de errores programados en él y busca a través de su código para encontrar los errores y los muestra en Eclipse. Los resultados son a veces discutibles, pero aún así es una herramienta bastante útil.

[Checkstyle](#): Esta herramienta hace que sea fácil para todos los que trabajan en un proyecto seguir las mismas reglas para la codificación. Esto hace que sea mucho más fácil de

entender el código de cada uno y el desarrollador también puede programar Checkstyle a seguir las reglas que les gusta.

[Hibernate](#): Cuando usted necesita convertir los datos en objetos Java, este es el mejor complemento disponible. Se entra en la base de datos SQL y regresa con todos los archivos de Java que necesitan ser convertidos en objetos Java. El plug-in reverse en una base de datos en cuestión de segundos.

[UML Designer](#): UML significa Unified Modeling Language y es un estándar para la construcción, y documentación de los elementos de los proyectos de programación orientada a objetos. También ayuda con la representación del árbol de XML.

[Ant Visualizer](#): Este visualizador analiza el XML y lo convierte en una serie de bloques, lo que le permite imaginar cómo avanza un proceso de construcción.

[Unnecessary Code Detector](#): Detector código innecesario, cuando se termina un proyecto de código, un programador a menudo se deja a la tarea de quitar el código innecesario que se ha puesto en él. Este plugin funciona a las mil maravillas en estos casos. Coloca banderitas con el fin de mostrar que los métodos en el código son innecesarias.

[JFormDesigner](#): Este plugin es útil si está creando aplicaciones de escritorio en Swing y portarlos a Internet utilizando el Google Web Toolkit. Esta es una herramienta de ayuda para dibujar la interfaz de usuario.

[Color Theme](#): : Este plugin tiene una serie de planes para colorear sus archivos y se pueden utilizar donde se quiera.

ACTIVIDAD

Instala los siguientes plugins utilizando para cada uso un método diferente:

1.- Instalación desde **Repositorio**. Utiliza la opción de instalar desde archivo

The UML2 Tools Downloads page

<http://www.eclipse.org/modeling/mdt/downloads/?project=uml2tools>

provides a 0.10 I-build that provides

<http://www.eclipse.org/downloads/download.php?file=/modeling/mdt/uml2tools/downloads/drops/0.10.0/I201103290512/mdt-uml2tools-Update-incubation-I201103290512.zip>

Download to your local disk, then DND it to the work with field of Window->Install New Software...

2.- Instalación desde market place

Instala el plugin **Eclipse color theme**, te permitirá configurar distintas combinaciones de colores para tu editor.

3.- Instalación manual

Utiliza esta opción para traducir el entorno de eclipse a español. Para ello debes localizar la dirección de descarga del repositorio que encontrarás en la página del proyecto Babel, para tu versión de eclipse concreta.

Proyecto Babel: <http://www.eclipse.org/babel/downloads.php>

The UML2 Tools Downloads page

<http://www.eclipse.org/modeling/mdt/downloads/?project=uml2tools>

provides a 0.10 I-build that provides

<http://www.eclipse.org/downloads/download.php?file=/modeling/mdt/uml2tools/downloads/drops/0.10.0/I201103290512/mdt-uml2tools-Update-incubation-I201103290512.zip>

Download to your local disk, then DND it to the work with field of Window->Install New Software...

WindowsBuilder

<http://www.eclipse.org/windowbuilder/download.php>

ArgoUML para Eclipse

ArgoEclipse is available as both an Eclipse plugin and as a standalone application with all the necessary Eclipse RCP components bundled.

PLUGIN INSTALLATION INSTRUCTIONS

The ArgoEclipse plugin requires Eclipse 3.5 or later. You can download Eclipse kits from the [Eclipse web site](#).

1. In the Eclipse menu go to *Help* → *Install New Software*
2. In the Install dialog click on *Add...* and add a site with the following attributes:
3. Name: ArgoEclipse
4. URL: http://argoeclipse.tigris.org/update_site
5. From here it's simple to follow the instructions from the wizard