

La empresa BK continua con el desarrollo de las interfaces para la aplicación de gestión hotelera y solicitan tu ayuda para completar las interfaces que necesitan. Entre otras, la empresa tiene la necesidad de cubrir la reserva de salones disponibles para convenciones, eventos, congresos, etc. La práctica que debes realizar consiste en crear una **interfaz que permita gestionar la reserva** de uno de estos salones, llamado **"Salón Habana"**. Los requisitos que debe cumplir esta interfaz son:

1. Se debe escribir el **nombre y teléfono** de contacto de la persona que hace la **reserva**.
2. Hay que cumplimentar:
 - ✓ **Fecha** del evento. Utiliza un **spinner** al que deberás modificar el modelo.
 - ✓ **Tipo**, se puede escoger entre **banquete, jornada o congreso**. Componente a tu criterio.
 - ✓ **Número de personas** que asistirán. Componente a tu criterio.
 - ✓ **Tipo de cocina** que se precisará, a elegir entre **bufé, carta, pedir cita con el chef o no precisa**. Componente a tu criterio.
3. Si el evento es un congreso se debe preguntar **el número de jornadas** del congreso y si se requerirán **habitaciones para los asistentes** al evento. Esto puedes hacerlo añadiendo los componentes necesarios con la **propiedad enabled a falso**.

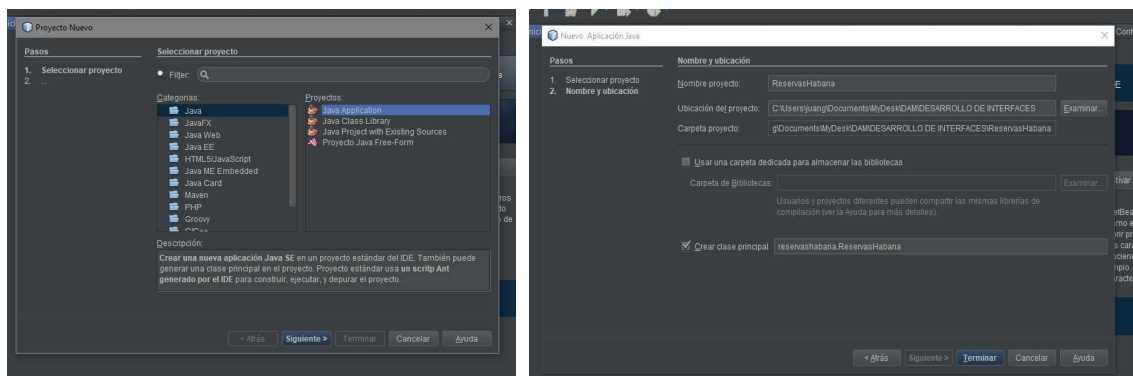
Tu tarea consistirá en generar una aplicación que cuente con un diálogo nuevo en el que crearás la interfaz con las restricciones que se indican más arriba. Además de añadir los componentes que necesites para darle funcionalidad deberás cumplir con los siguientes **requisitos no funcionales**:

- ✓ Debes **cambiar el nombre y añadir un ToolTipText** a los componentes.
- ✓ **Modifica el formato de texto** de las etiquetas y **añade separadores y paneles con título** para aportar vistosidad a la interfaz.
- ✓ Utiliza el modo de diseño libre para colocar los componentes en el diálogo. Cuida que la **composición quede armónica**.
- ✓ La **interfaz generada** irá **conectada a la interfaz principal** de la aplicación **mediante el menú principal y mediante un botón** que debes añadirle que tendrás que programar correctamente **para que abran el diálogo** que has creado.
- ✓ Debes **crear la interfaz modal**.
- ✓ Cuando se selecciona como tipo de **evento banquete o jornada** los componentes para **número de días y habitaciones** deben estar **desactivados**, sin embargo si es congreso se deben activar.

1. Creación del proyecto en NetBeans

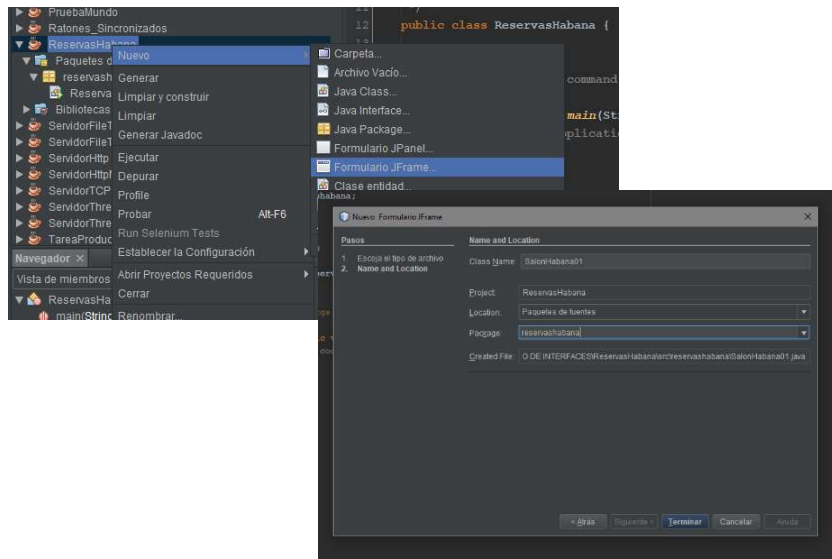
Para la realización de esta tarea, he creado un proyecto, siguiendo los pasos facilitados desde el tema.

Proyecto Nuevo/Java/Java Application y lo he llamado **ReservasHabana**



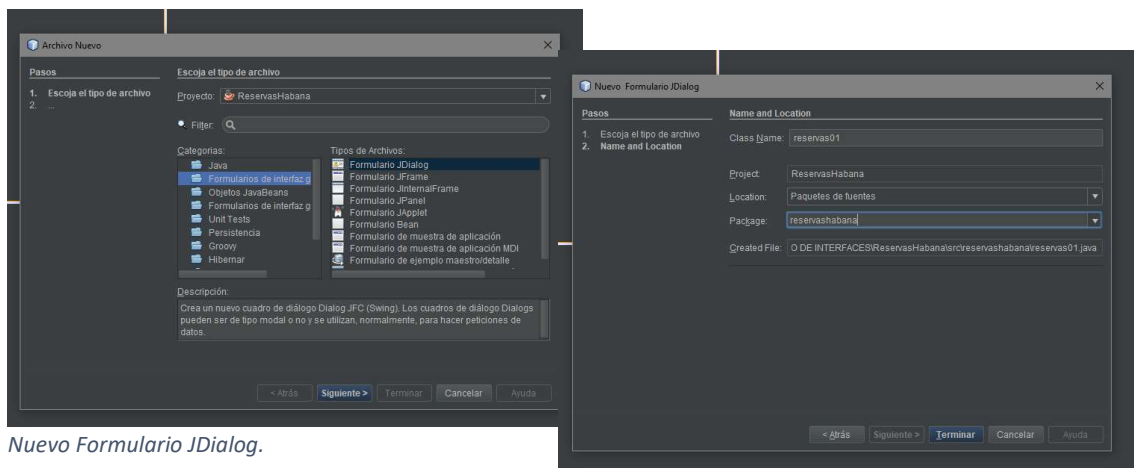
Creación de proyecto Java en NetBeans.

He creado primero la clase **SalonHabana01**, como un formulario **JFrame**, dentro del **package** del proyecto.



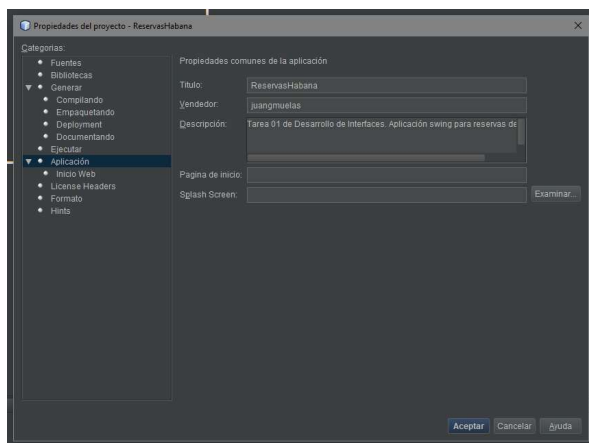
Nuevo Formulario JFrame.

A continuación un formulario **JDialog** para la clase **Reservas01**, que gestiona la información a recibir.



Nuevo Formulario JDialog.

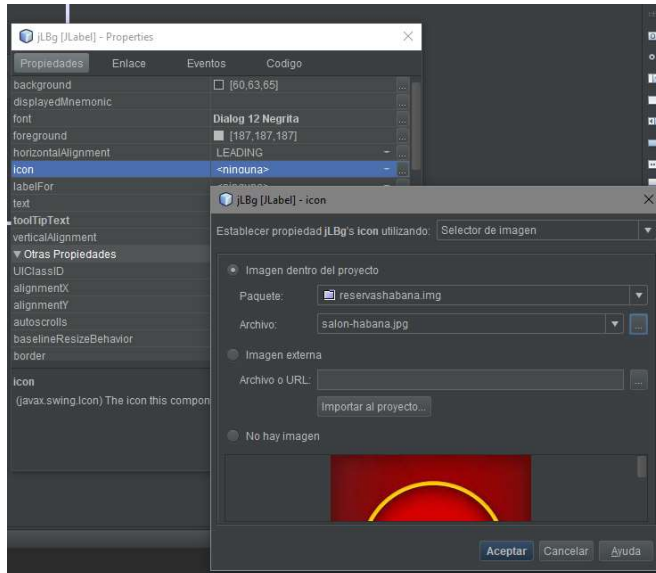
He añadido también las Propiedades comunes de la aplicación.



Descripción de Propiedades comunes del proyecto.

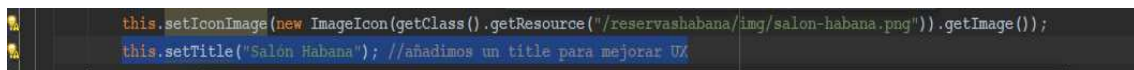
2. Desarrollo de las interfaces.

La primera parte del desarrollo de la aplicación es meramente visual, con el fin de mejorar la experiencia de usuario, añadiendo en el **JFrame** el logo del Salón Habana, mediante la inserción de un **JLabel** al que le he modificado la propiedad **icon** para añadirle la imagen.



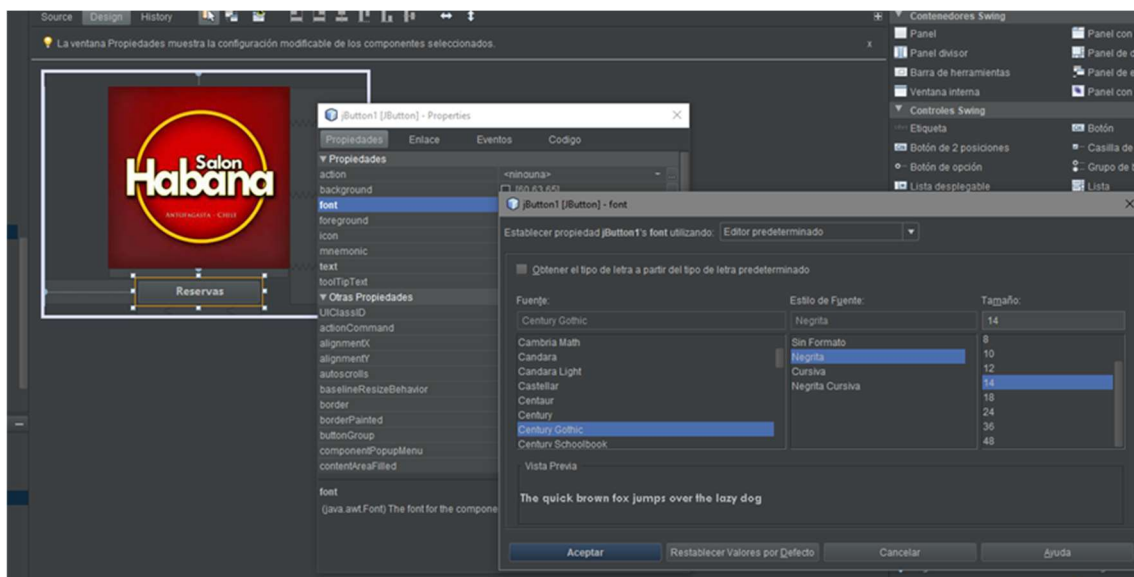
Añadiendo logo de imagen.

A continuación, he utilizado también el logo para que aparezca como **favicon** de nuestra aplicación y añadido un **title** identificativo a su lado, añadiendo un par de líneas de código tras iniciar el componente.



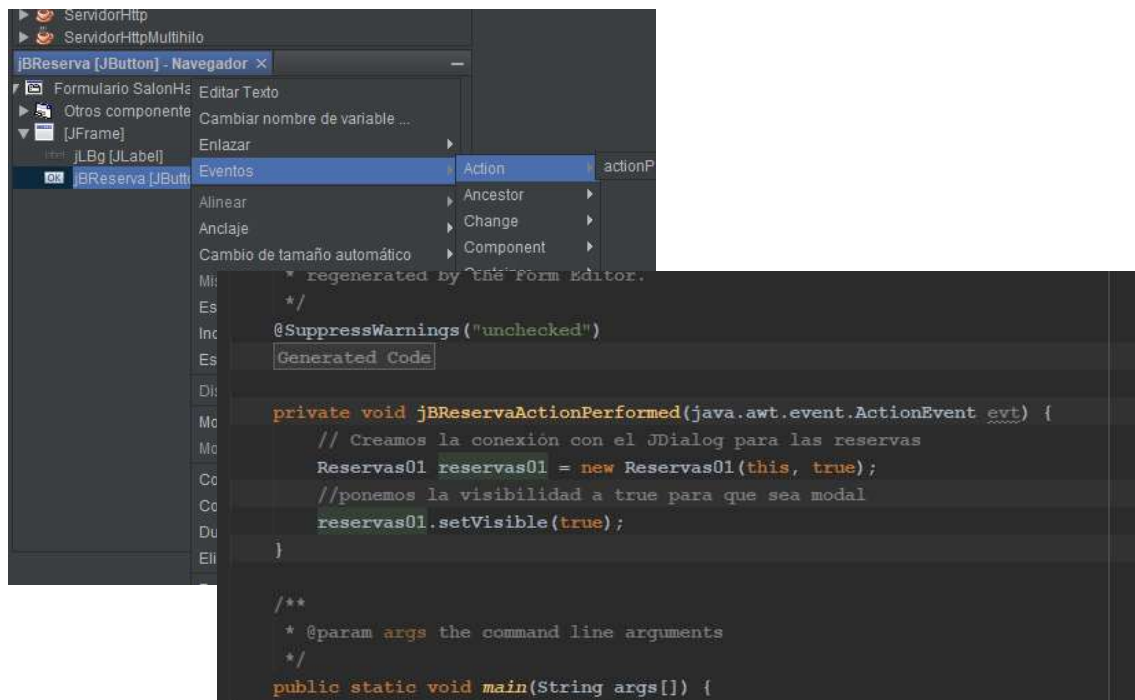
Añadiendo el title a nuestra ventana de proyecto.

Comenzamos a darle funcionalidades insertando un elemento **JButton**, aprovechando en este momento para modificar aspectos como la tipografía, mediante **Century gothic**, más parecida a la del logo, y modificando su propiedad **background** y añadirle un rojo similar al exterior del logo.



Modificación de fuentes en las propiedades del objeto.

Generamos el evento de nuestro botón para que se comunice de forma modal con el **JDialog Reservas01**



Evento para acceder a nuestro JDialog.

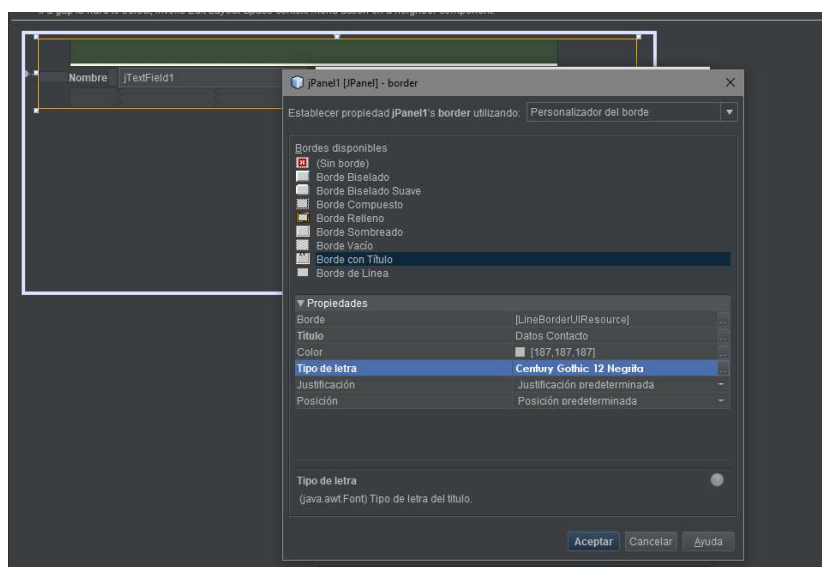
Pasamos a diseñar el área de reservas en el **JDialog**.

Se añaden un **JPanel**, al que se modifican las propiedades para añadirle un borde con título.

A este, se le modifica además la tipografía para unificarla en toda la aplicación.

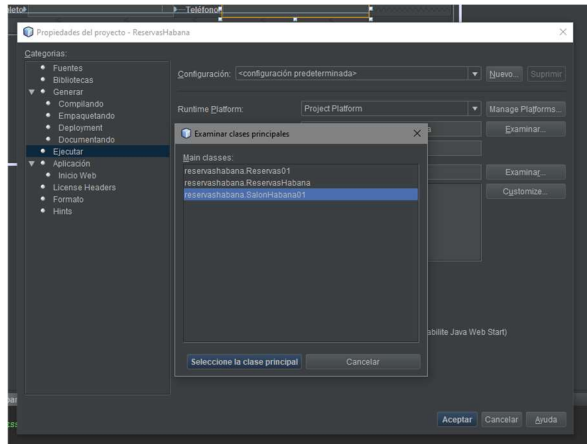
También se añaden dos **JLabel** y dos componentes **JTextField** que recogerán el nombre y el teléfono de la persona que realiza la reserva.

Se aprovecha para ir modificando el nombre de las variables, su texto o los **tooltiptext** para mejorar la usabilidad.



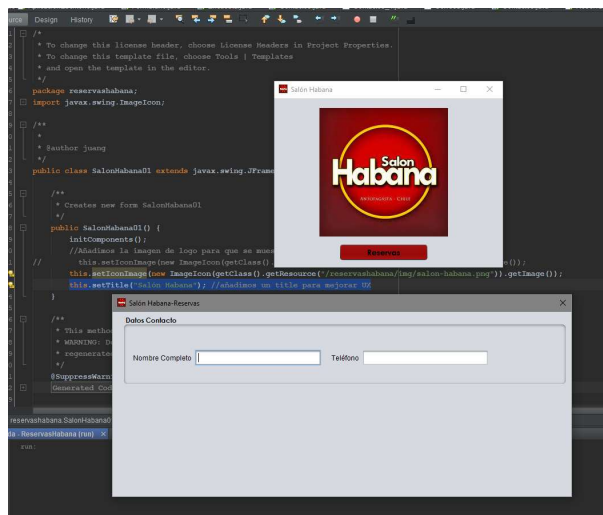
Personalizaciones y mejoras de usabilidad.

Antes de seguir, comprobamos el correcto acceso a esta parte, poniendo la clase anterior como principal.



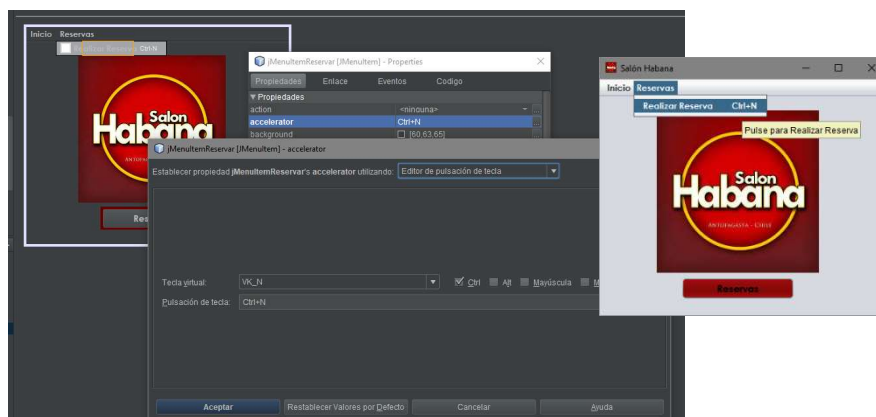
Seleccionando clase principal.

Ejecutamos la aplicación y comprobamos que vamos por el camino correcto.



Prueba de ejecución.

Aunque así ya sería plenamente funcional, la tarea nos pide añadir un acceso a través de un menú, por lo que se añade, mediante la inserción de un **JMenuBar**, al que se modifica sus elementos, llamados ahora Inicio (sin servicio, pero que podría llevar a una aplicación principal) y Reservas, con un **JMenuItem**, al que además de modificar, unificando los criterios de usabilidad, se le mejora la accesibilidad mediante un atajo de teclado **Ctrl+N**

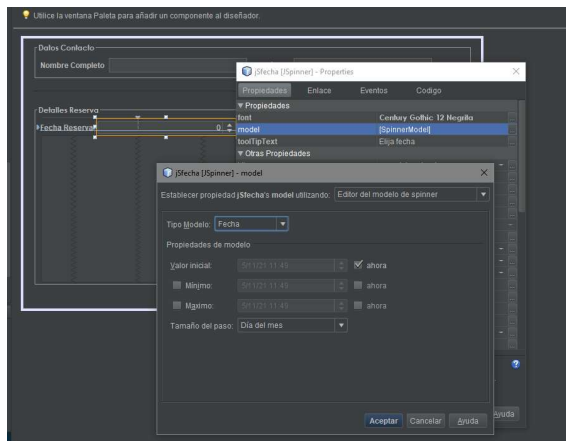


Mejora de accesibilidad mediante atajos de teclado.

Se le añade el evento correspondiente para poder acceder al **JDialog** de **Reservas01**.

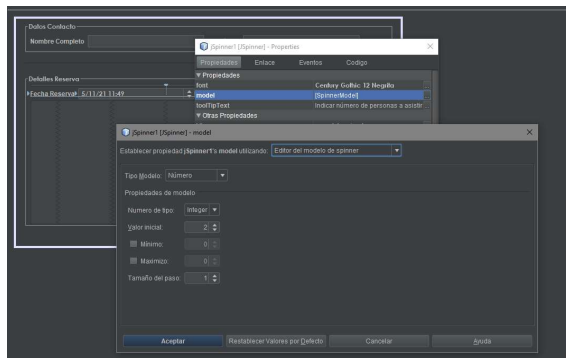
```
private void jMenuItemReservarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new Reservas01(this, true).setVisible(true);
}
```

Proseguimos con el diseño, diferenciando la parte de los detalles de la reserva, que habremos separado previamente con un Contenedor **JSeparator**, antes de añadir un nuevo **JPanel**, con propiedades similares al anterior, y donde añadimos primeramente el **JLabel** que acompañará visualmente a un **Spinner** de tipo fecha, modificados también con los parámetros comentados anteriormente de tipografía y usabilidad.



Propiedades de *JSpinner* de tipo date.

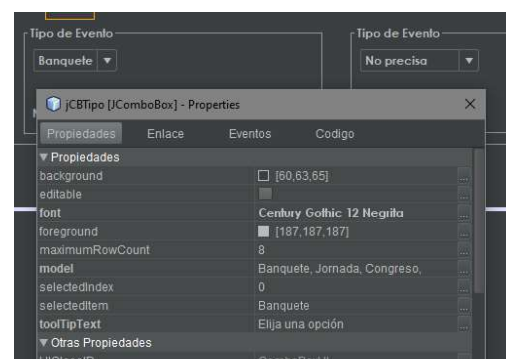
Para los asistentes, prácticamente es duplicar el paso anterior, sólo que debe ser un **spinner** numérico y le vamos a facilitar un valor inicial de '2', por suponerse que son experiencias grupales.



Propiedades de *JSpinner* de tipo date.

Dentro del **JPanel** actual, volvemos a crear una división donde añadimos los tipos de evento y de cocina, a través de dos **JPanel** de características similares a los anteriores.

Dentro de ellos, añadimos un Control **Swing** de tipo **ComboBox** o Lista desplegable, con los distintos tipos de evento/cocina, y modificamos el resto de las propiedades o nombres de variables, siguiendo las pautas previas.

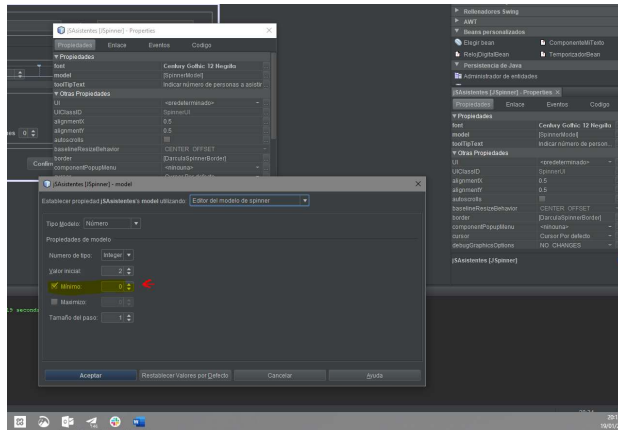


Añadir elementos de lista.

En la tarea, se nos pide que en el caso de ser un congreso, se añadan los campos para detallar el número de días y posible reserva de habitaciones.

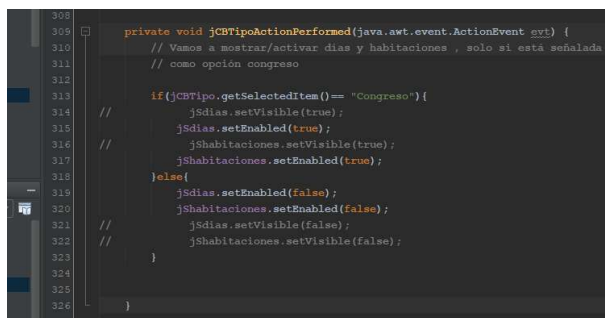
Los creamos duplicando el **spinner** numérico anterior, modificando obviamente todas sus propiedades y los valores iniciales, que serán de '1' jornada y '0' habitaciones.

Me aseguro de que los valores de los **spinner** no sean negativos:



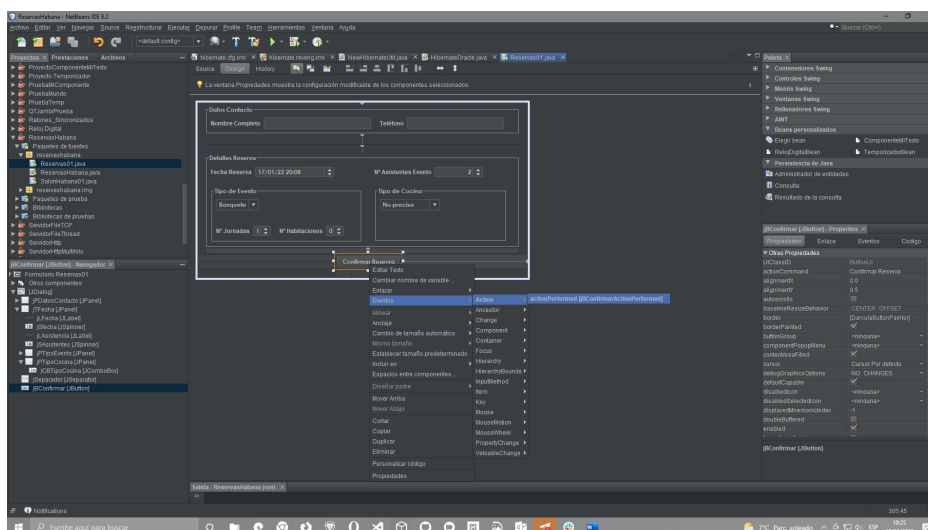
Ajustando valores de nuestros Spinner.

Para que solo puedan seleccionarse si se elige la opción “Congreso”, creamos un nuevo evento en este **ComboBox**, y vamos a su código para darle la visibilidad solo en ese supuesto, mediante un condicional.



Condición para activar nuestras opciones.

Para finalizar, mejoramos el resultado de la aplicación mediante un botón que nos muestre el resultado satisfactorio de nuestra reserva.



Evento para mostrar resultados (1).

Evento para mostrar resultados (2).

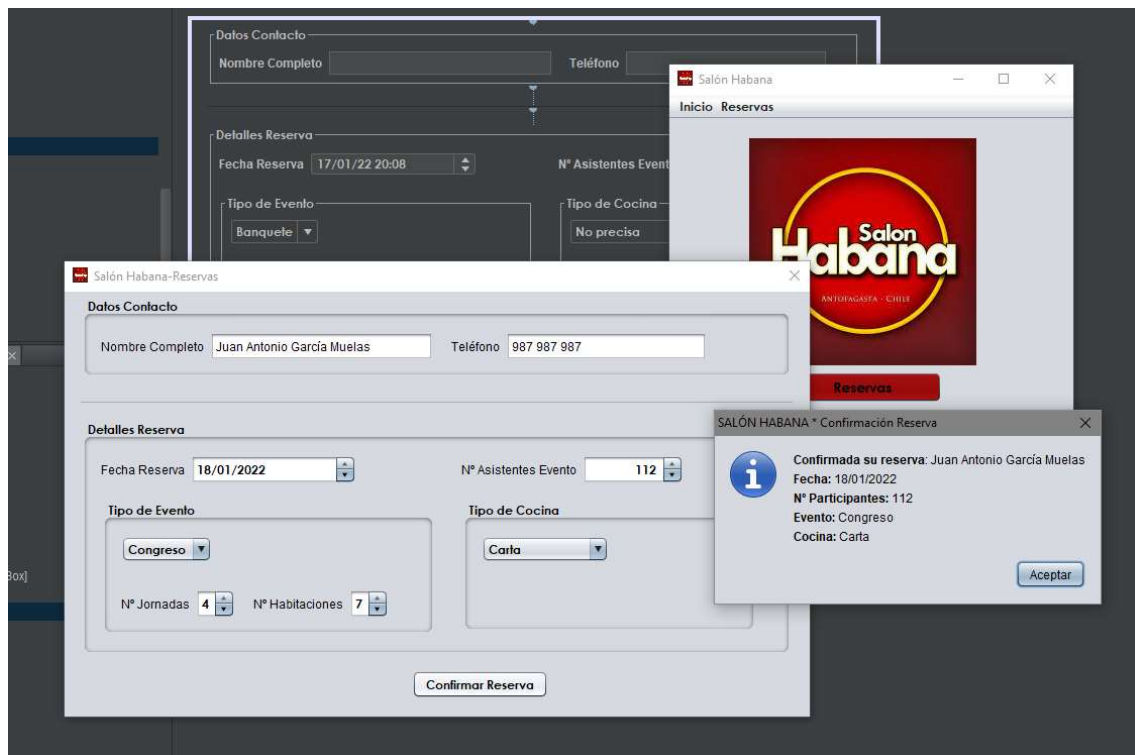
Ejecutamos para ver el resultado:



Probamos seleccionando Congreso:



Funciona correctamente, así que pulsamos sobre el botón de confirmación, para ver si nos muestra los datos:



Capturas de los procesos ejecutados y mensaje de confirmación.