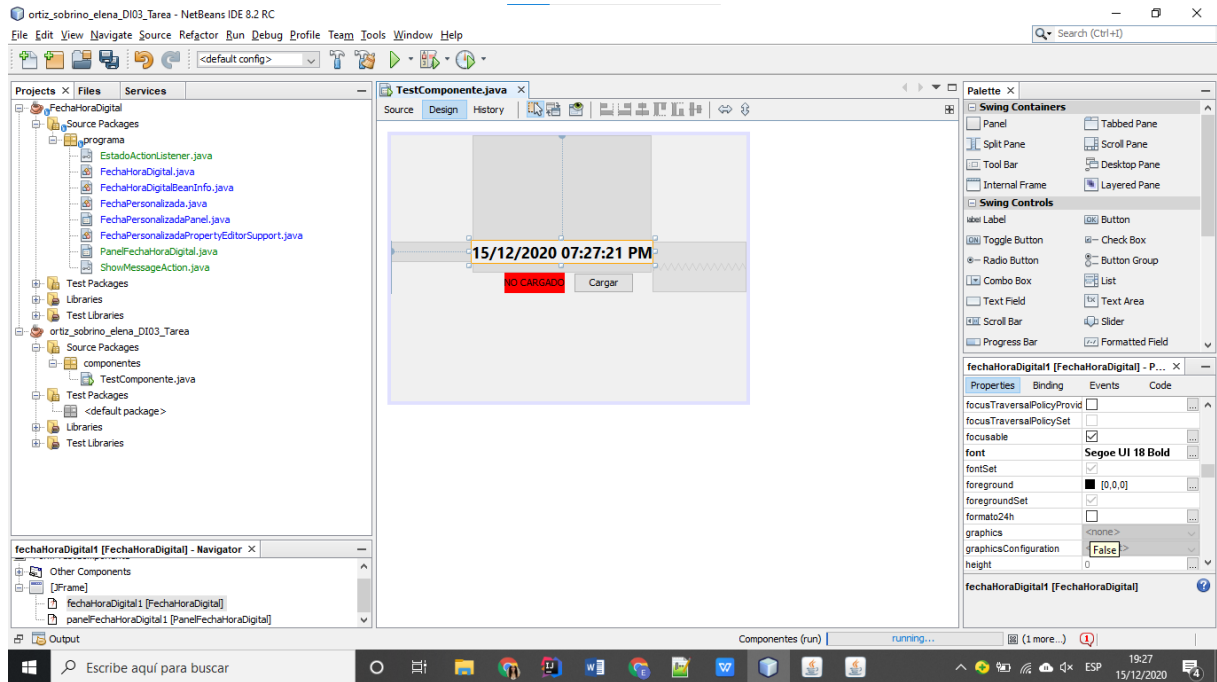


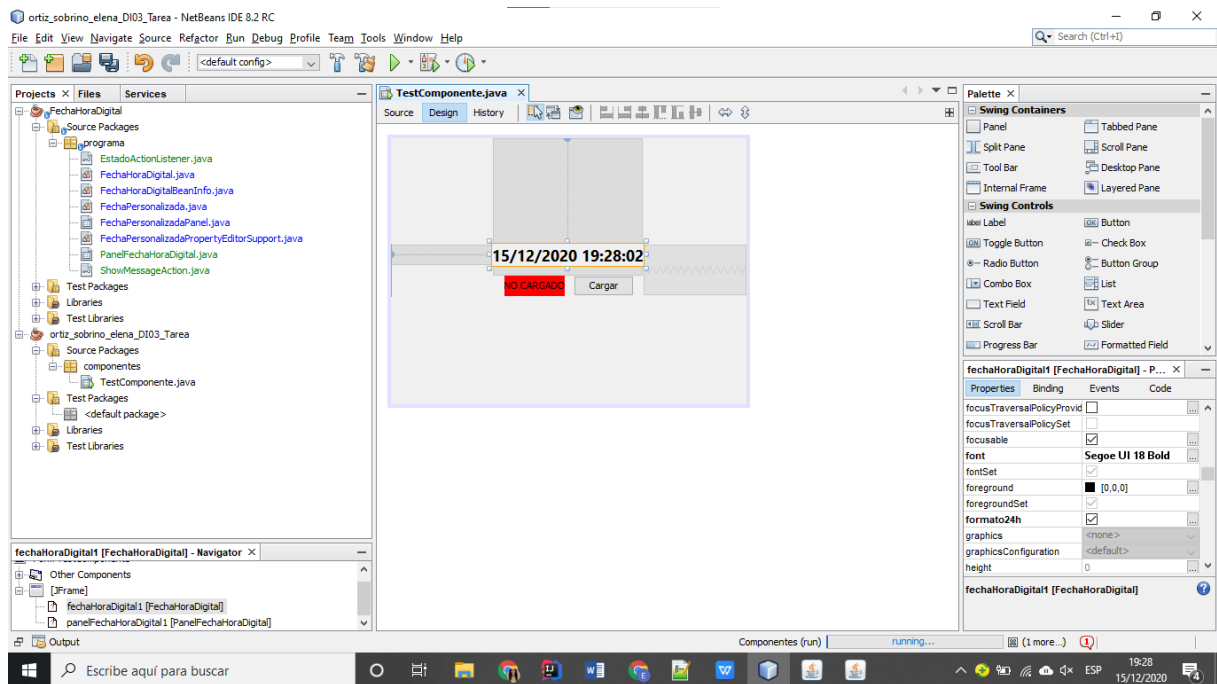
TAREA 3 – DESARROLLO DE INTERFACES

Componente Fecha y hora Digital

Formato 12 horas

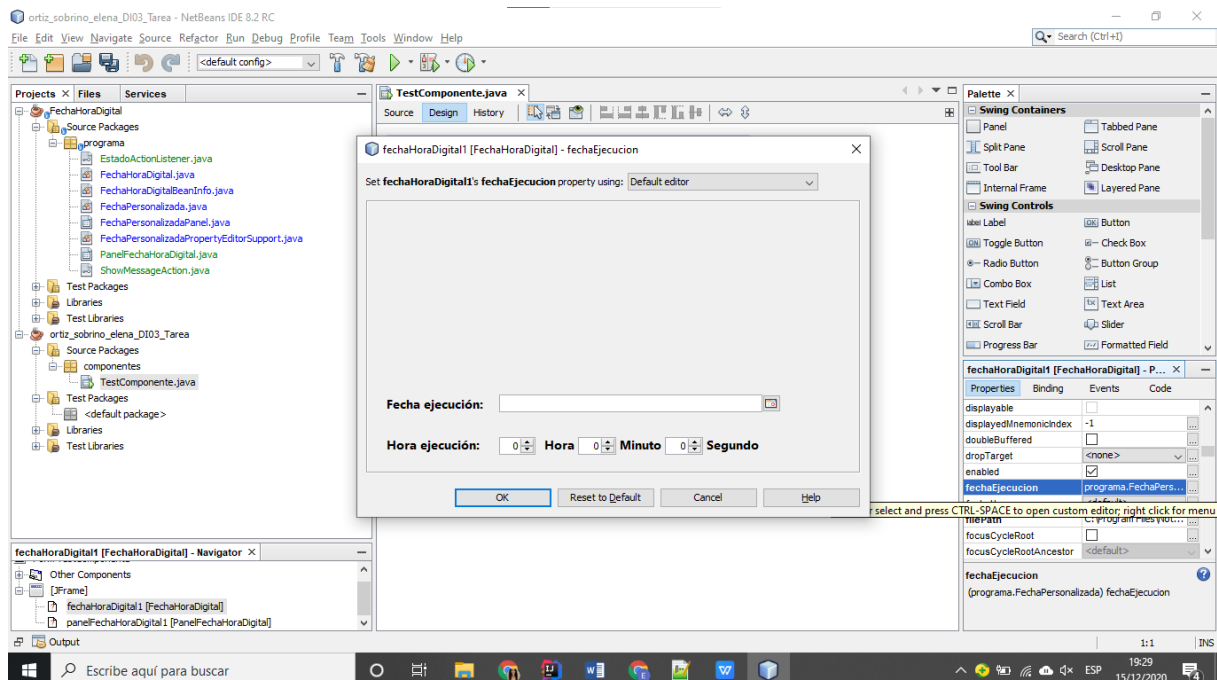


Formato 24 horas

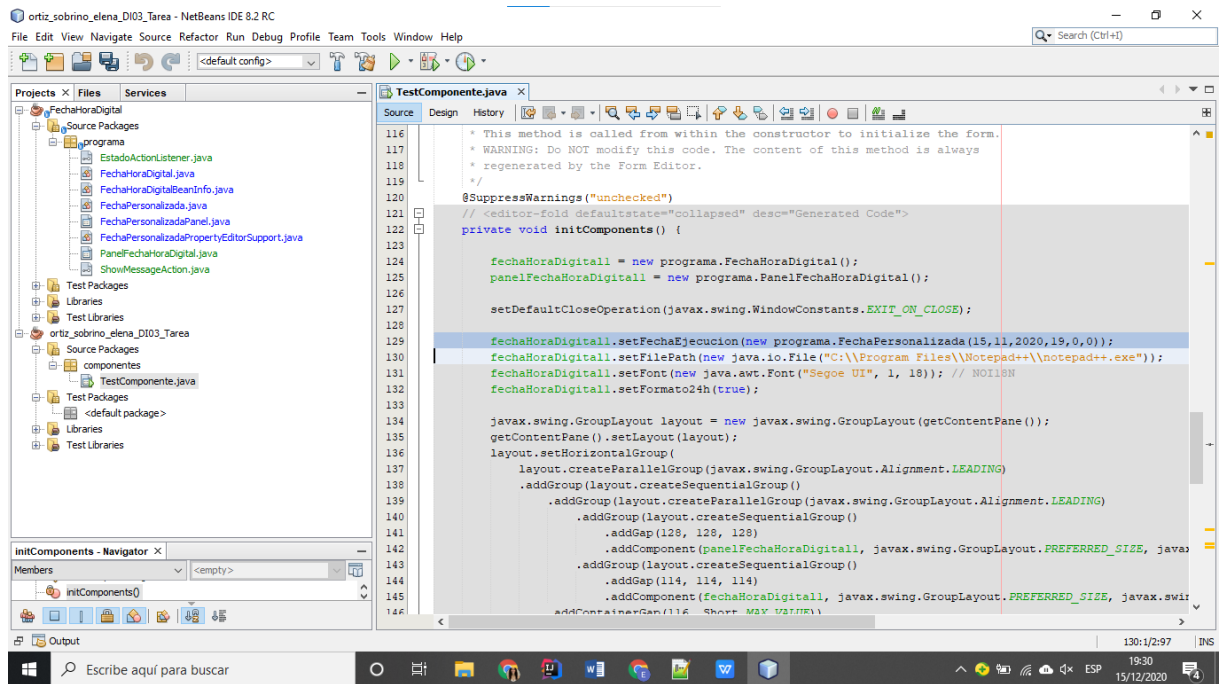


Fecha de ejecución

Se ha creado un componente personalizado donde se utiliza un `JDateChooser` para seleccionar la fecha y un spinner para seleccionar la hora

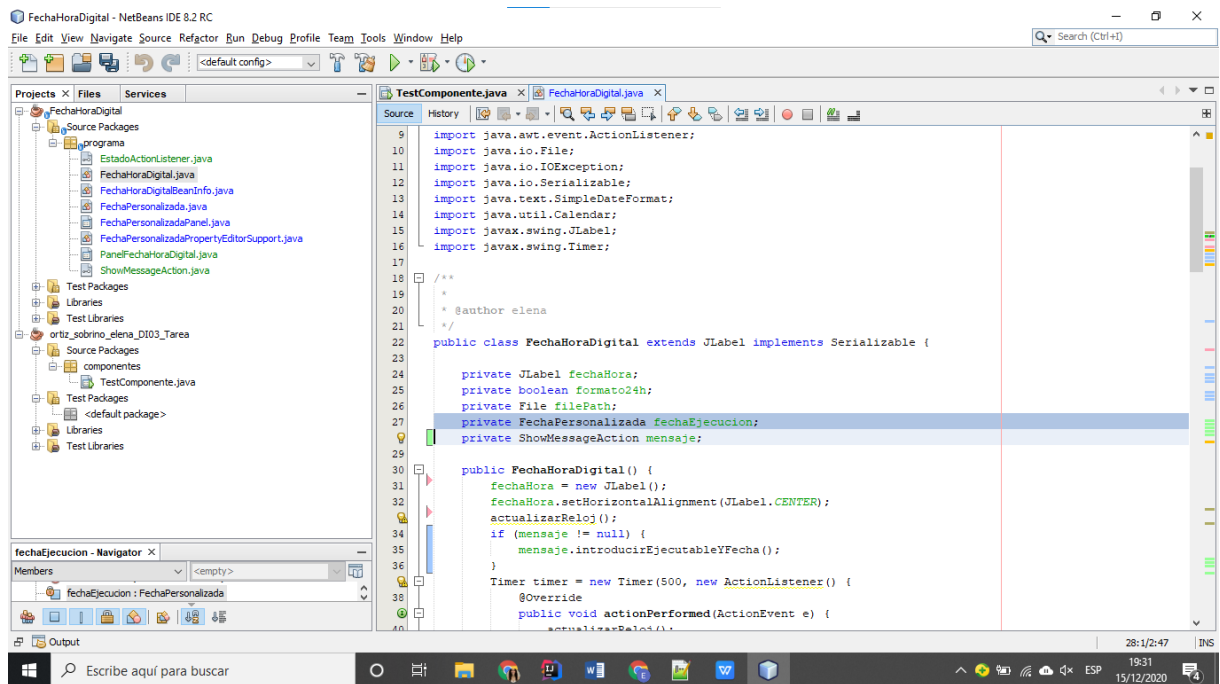


Los datos introducidos en esta ventana se asignarán automáticamente en el componente

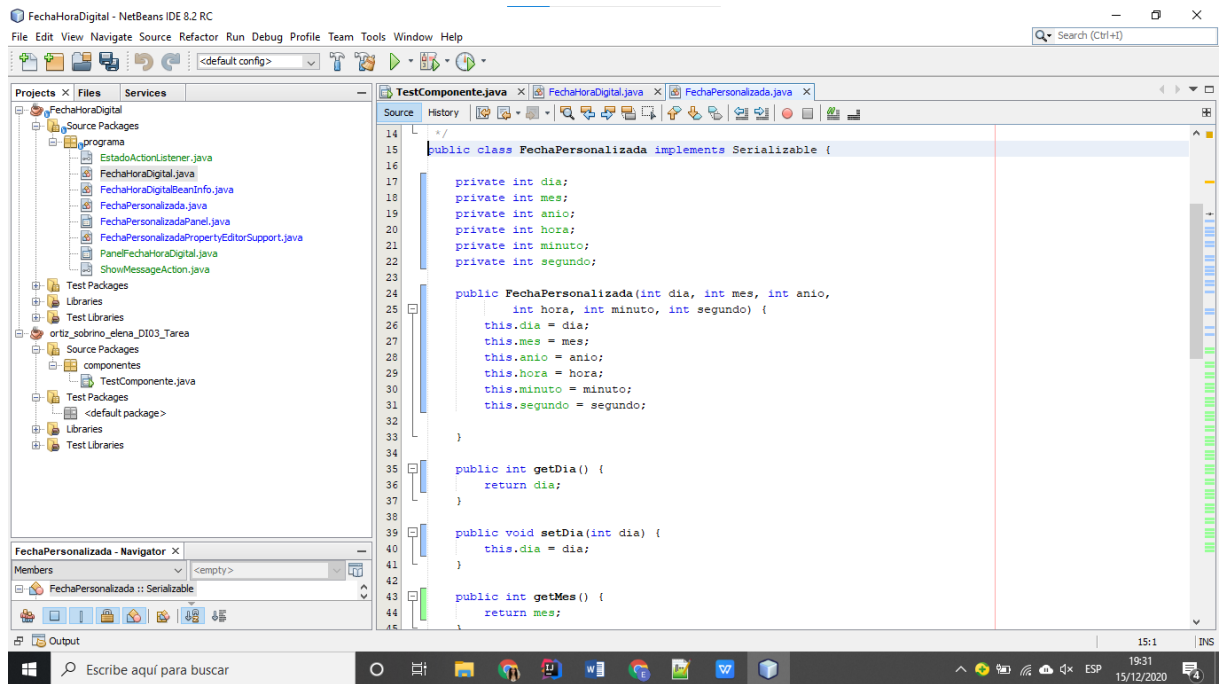


Para crear esta propiedad personalizada se hace lo siguiente

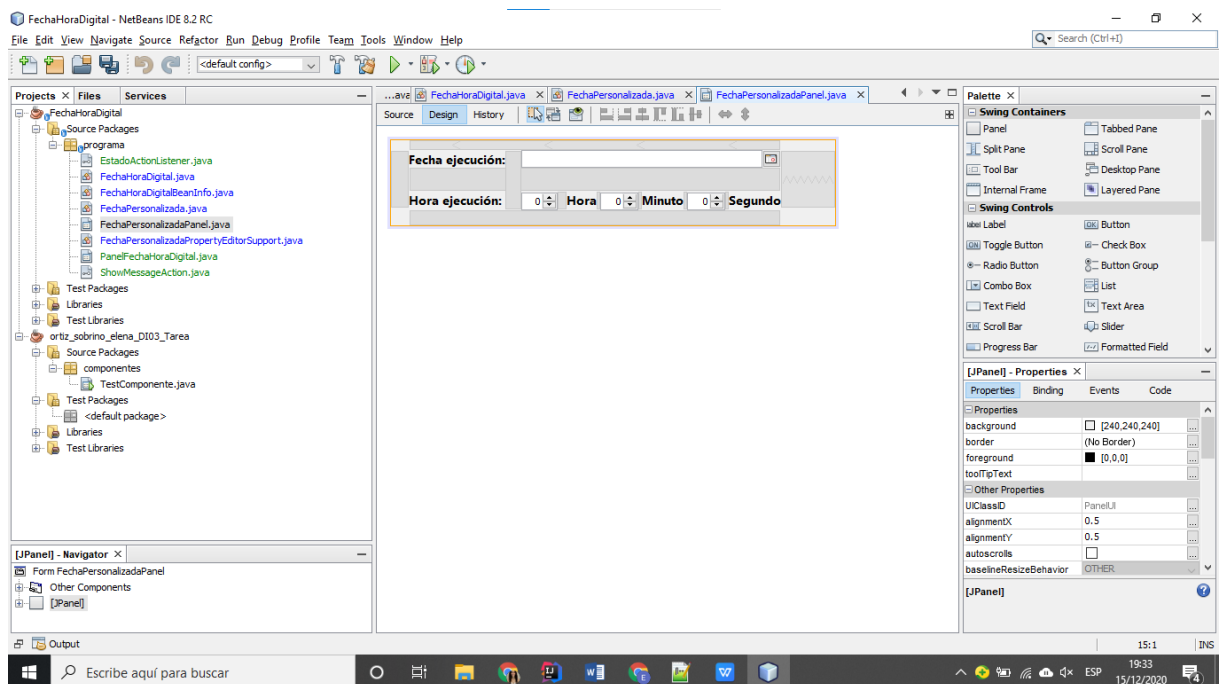
En el componente FechaHoraDigital creamos un atributo que es FechaPersonalizada



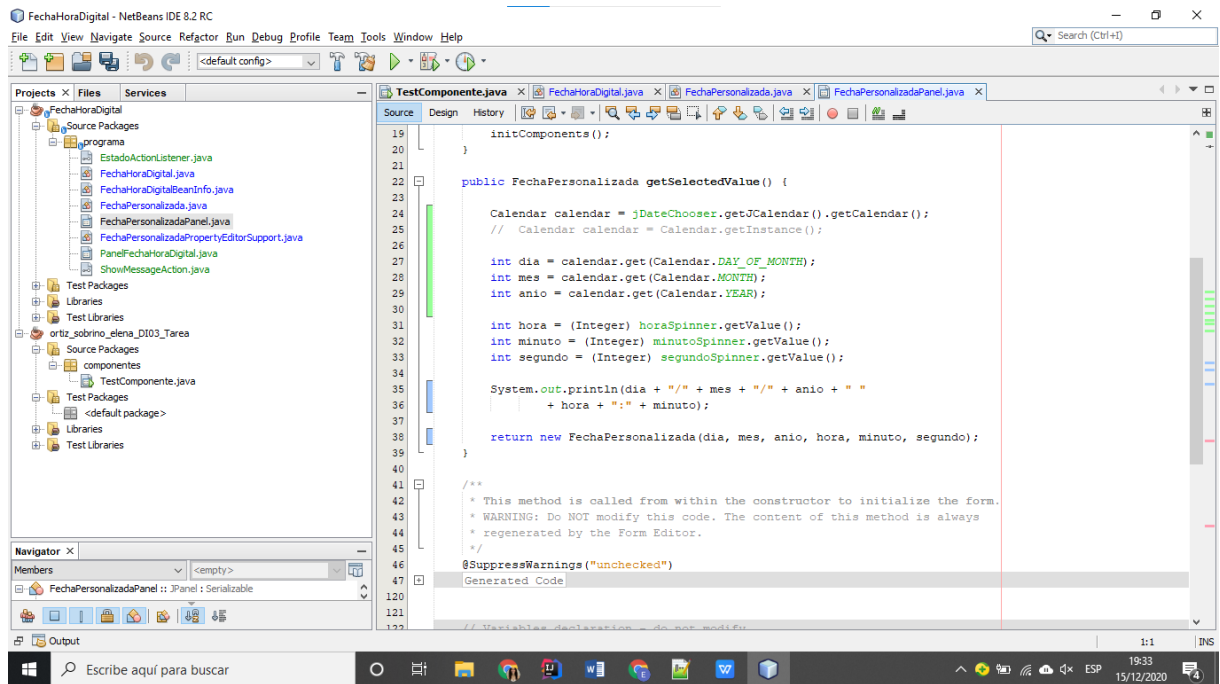
La clase FechaPersonalizada solo contendrá los atributos necesarios para crear una fecha y hora que sería el año, mes, día, hora, minuto y segundo y los getter y setter. Además deberá implementar serializable para poder hacer la configuración



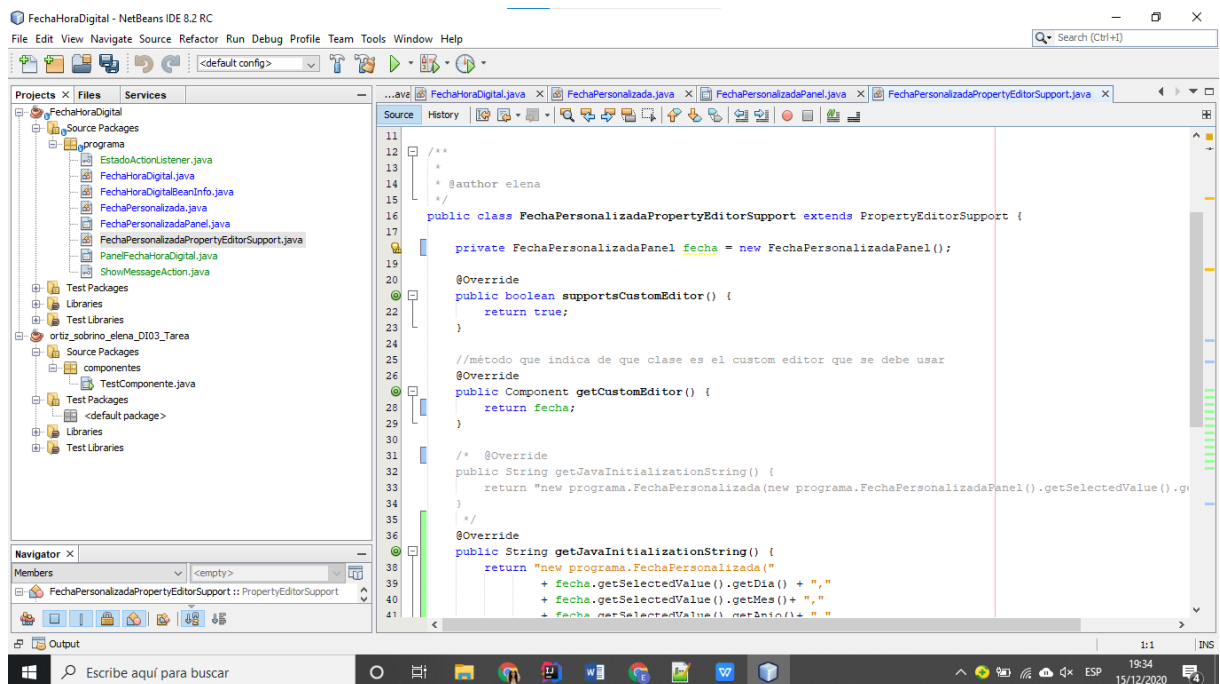
Creamos una clase FechaPersonalizadaPanel que será el diseño que mostraremos y nos servirá para recoger los valores que introduce el usuario



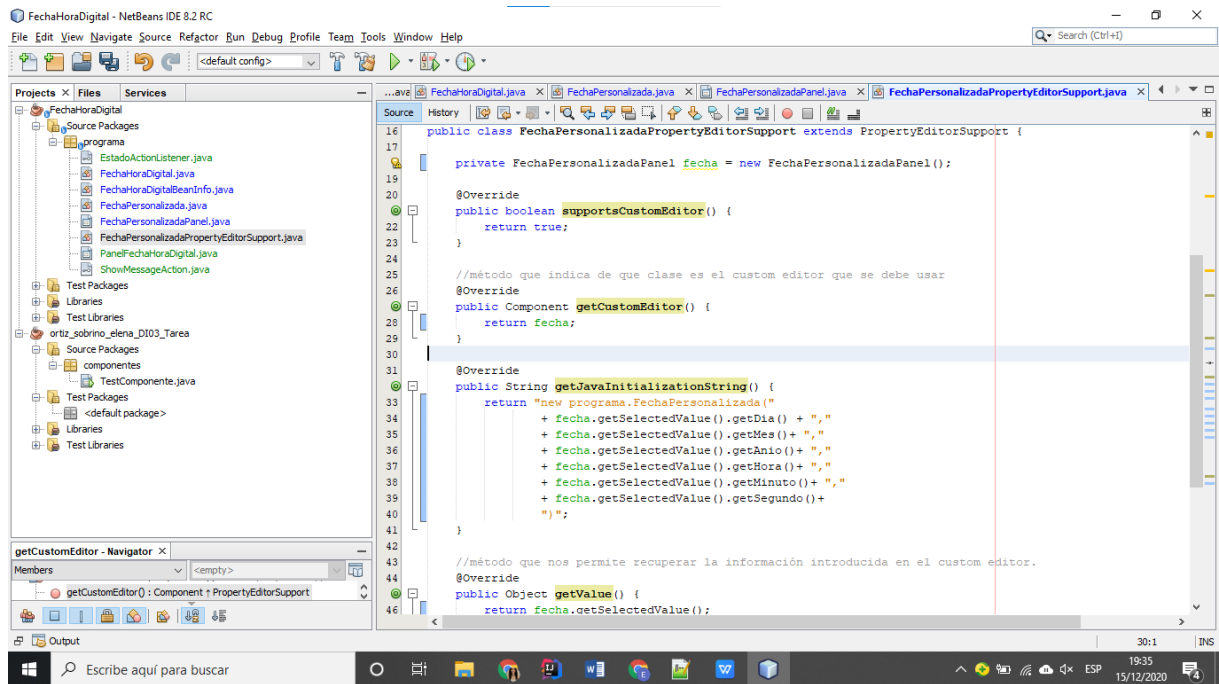
Estos valores se recogen a través de getSelectedValue



Creamos la clase FechaPersonalizadaPropertyEditorSupport que extienda de PropertyEditorSupport para que en el editor de propiedades para el campo FechaPersonalizada netbeans sepa de dónde tiene que sacar los datos para rellenar ese campo

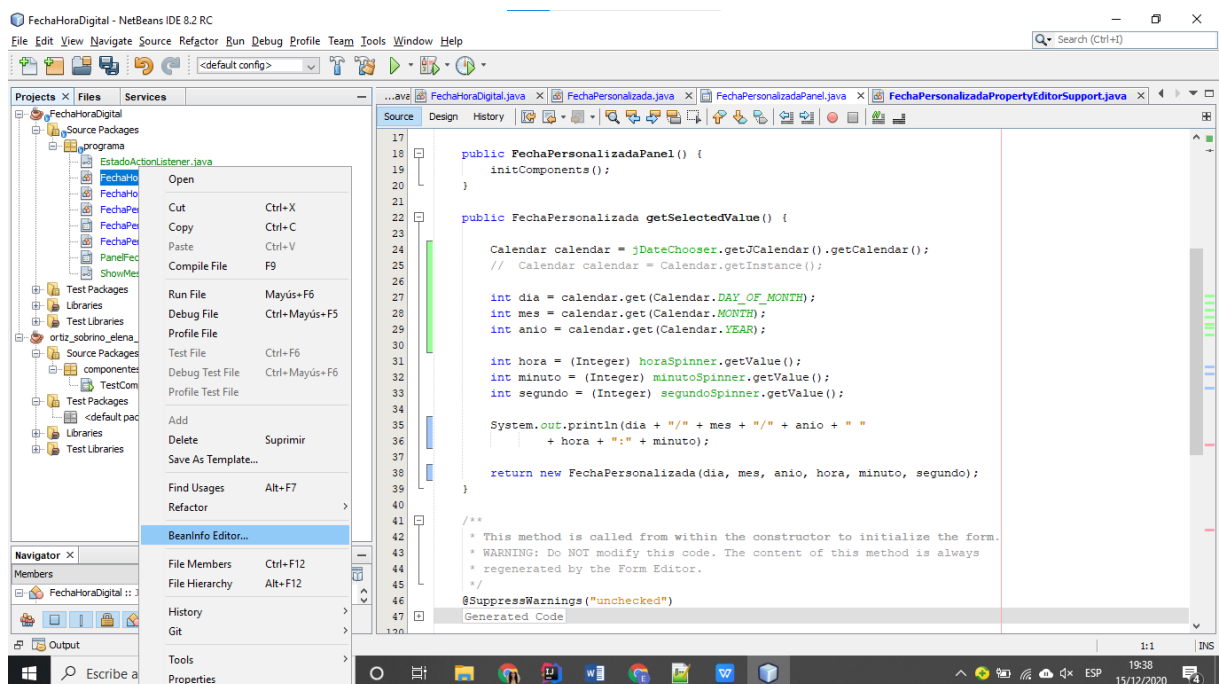


Para esto tenemos que sobrescribir los siguientes métodos

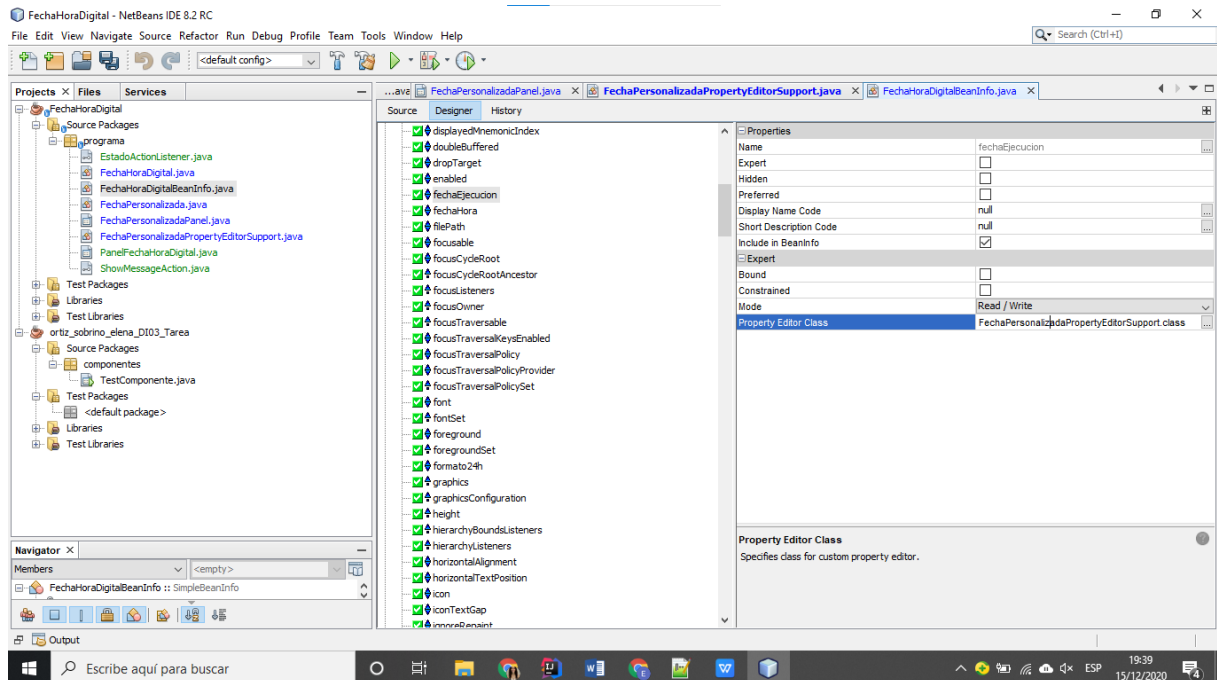


1. SupportsCustomEditor -> devolvemos true para decirle que utilizaremos una configuración personalizada
2. getCustomEditor() le informamos cuál será el panel del editor
3. getJavaInitializationString() será la forma de decirle al programa el valor que introduce el usuario
4. getValue() devolverá una nueva instancia de FechaPersonalizada con los datos que ha introducido el usuario

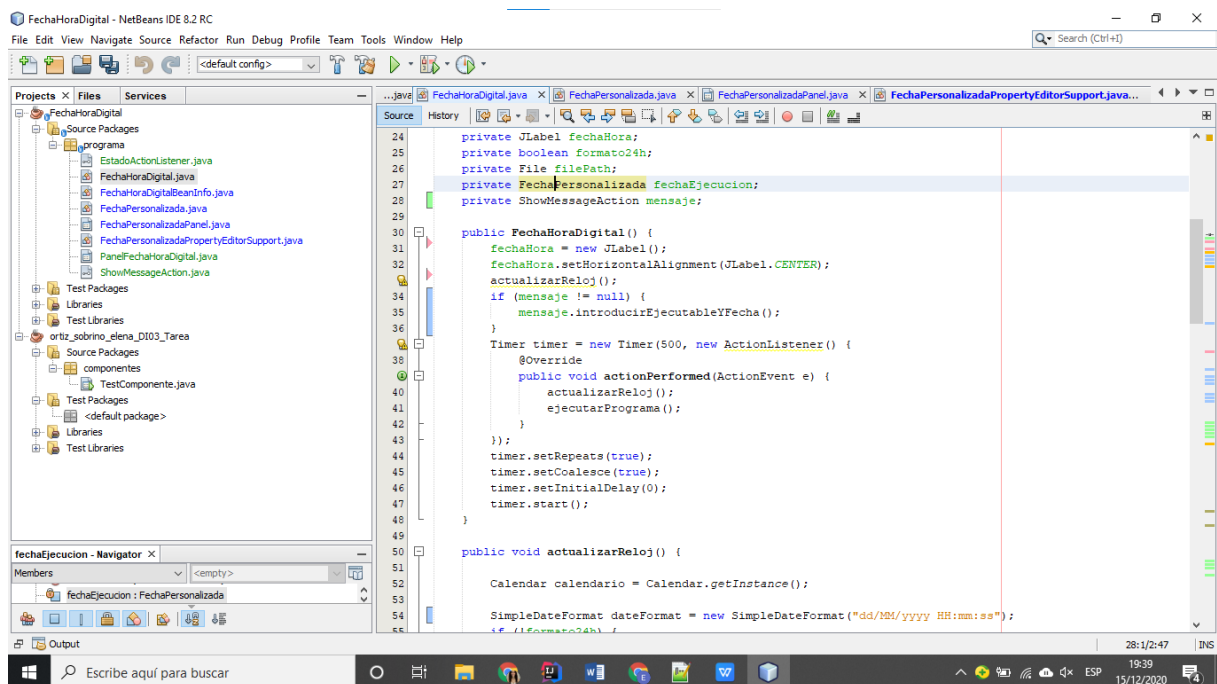
Creamos el BeanInfoEditor



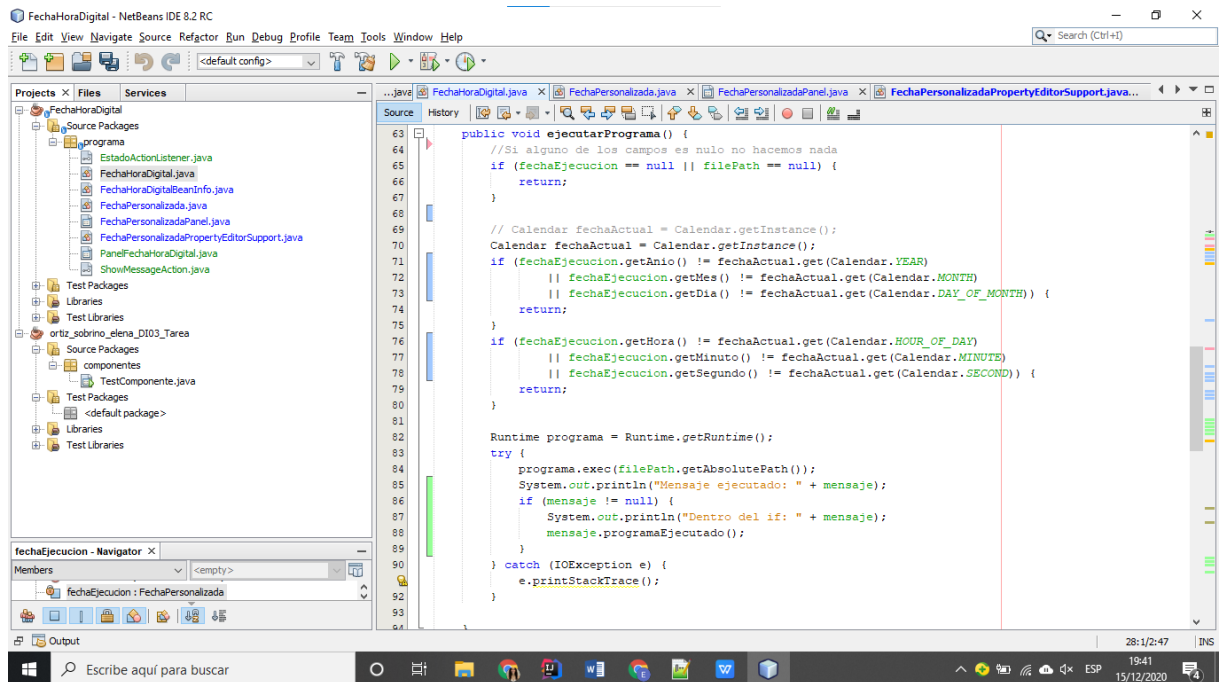
Nos vamos a esa clase y en la pestaña diseño le decimos dónde se encuentran las configuración de la propiedad



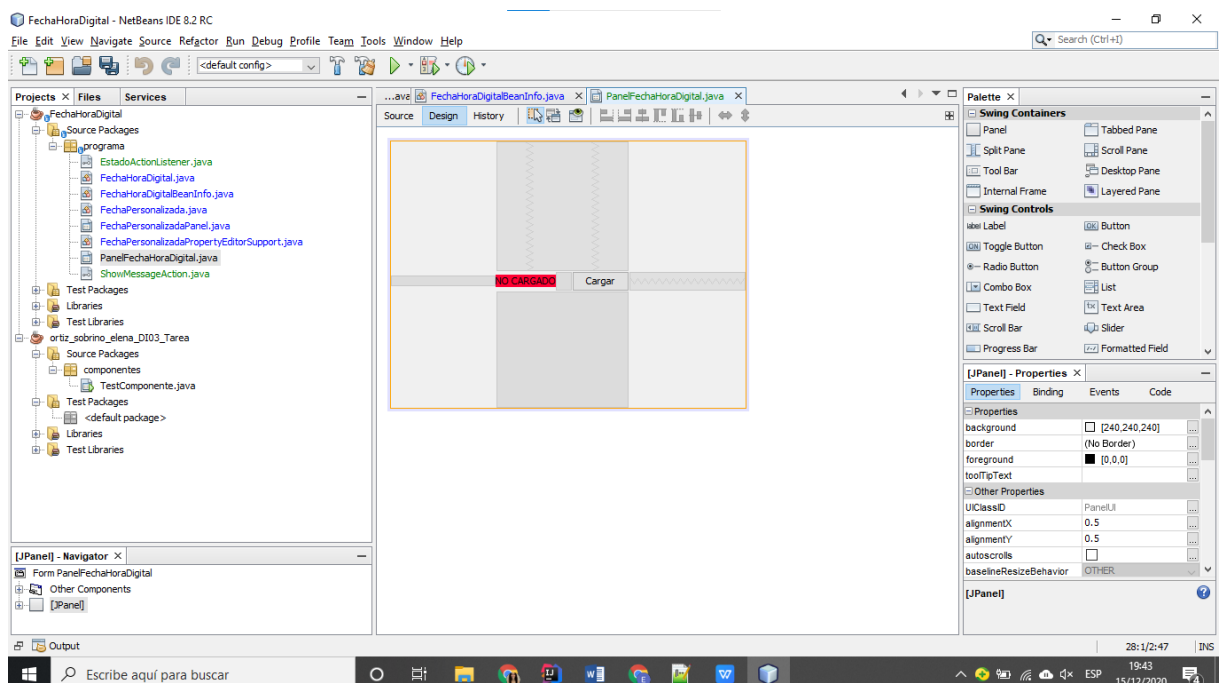
La clase FechaHoraDigital incluye los métodos para actualizar la hora cada segundo y para ejecutar el programa. Dentro del método que actualiza la hora mostramos el formato 24 horas o 12 horas dependiendo de si el usuario ha marcado la casilla



El método ejecutar programa comprueba que la fecha de ejecución del programa coincide con la que tiene que ser ejecutado y si no es así no hace nada. Cuando llega la fecha ejecuta el archivo que le hemos indicado en el filepath.



PanelFechaHoraDigital

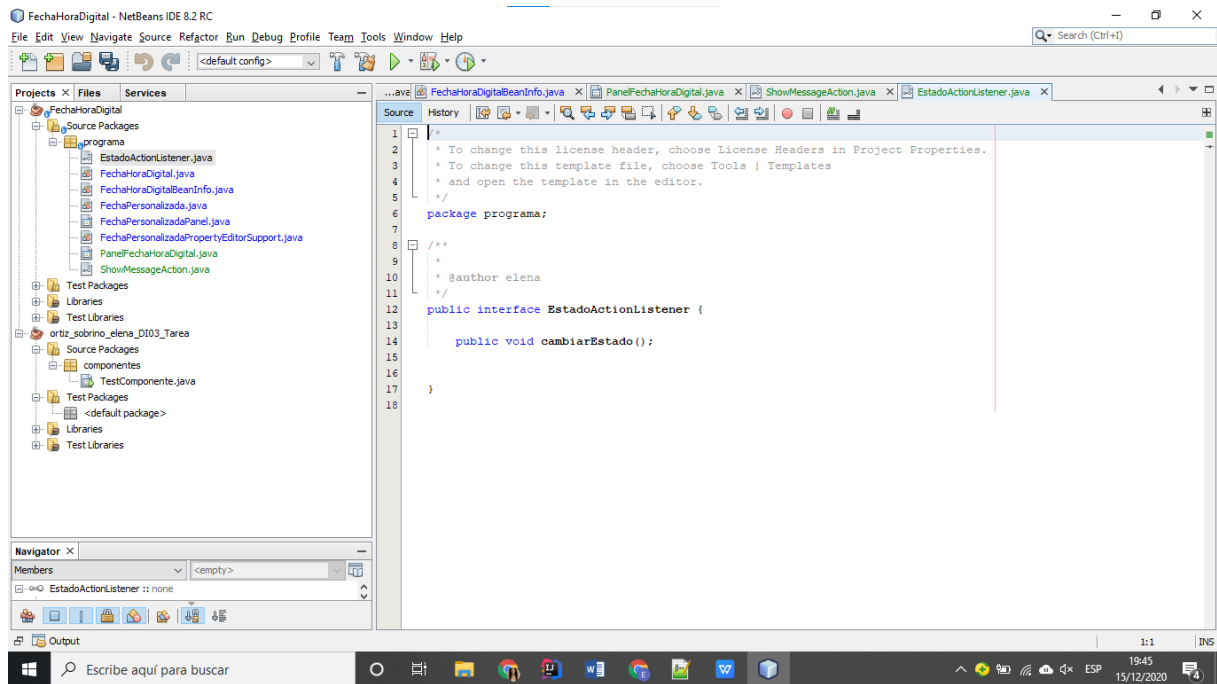


Será la segunda parte de nuestro componente que se compone de una etiqueta con los estados y un botón de cargar para modificar los estados

Interfaces

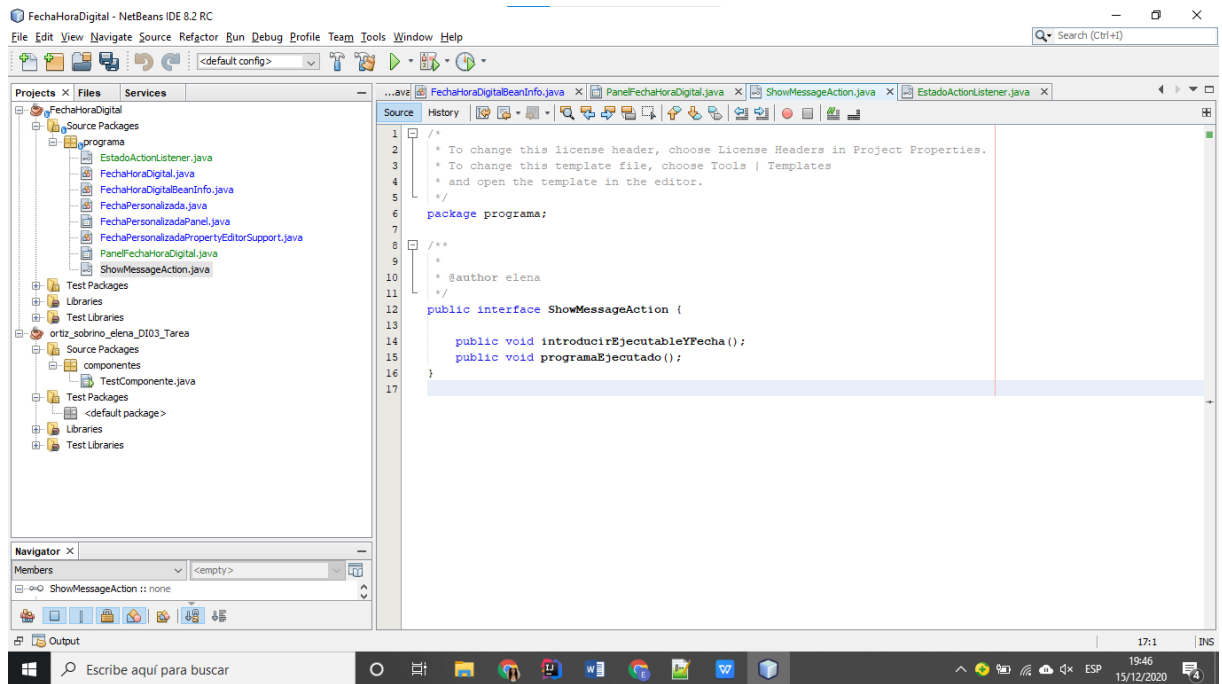
Como hay comportamientos cuya implementación puede variar en función del usuario que utilice el componente se crean dos interfaces cuyos métodos implementará el usuario

EstadoActionListener



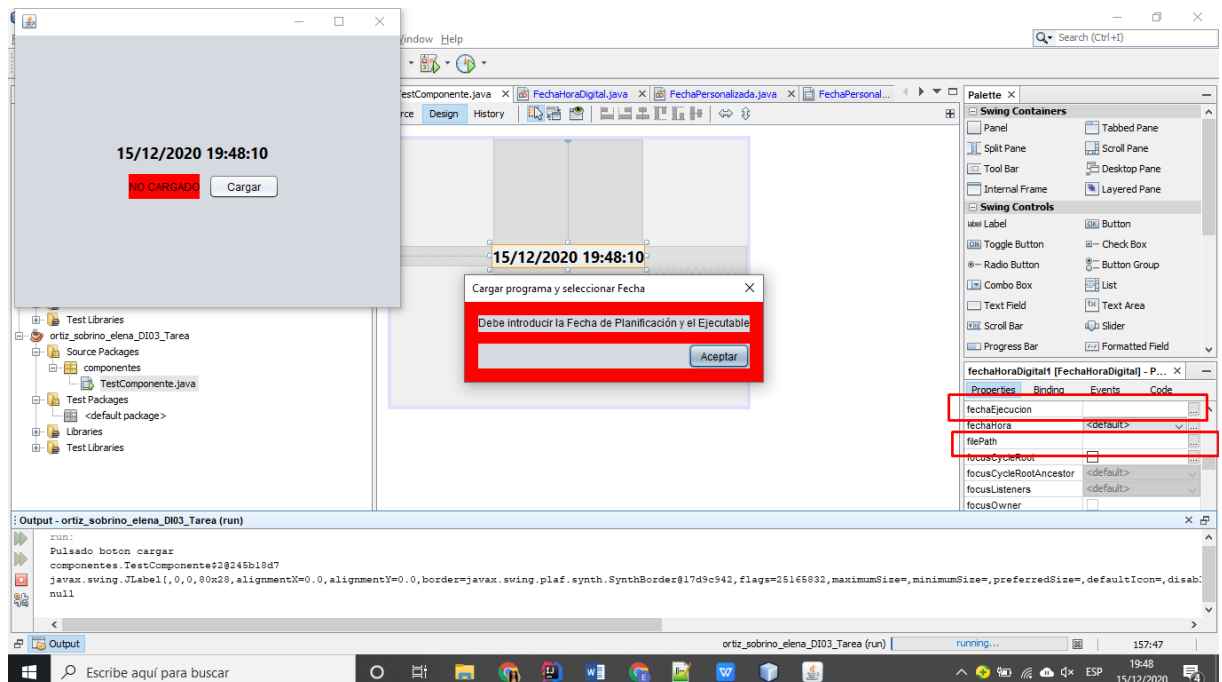
ShowMessageAction

Para que el usuario muestre los mensajes una vez ejecutado el programa o cuando el usuario no ha rellenado los atributos filePath o fechaEjecución necesarios para ejecutar el programa

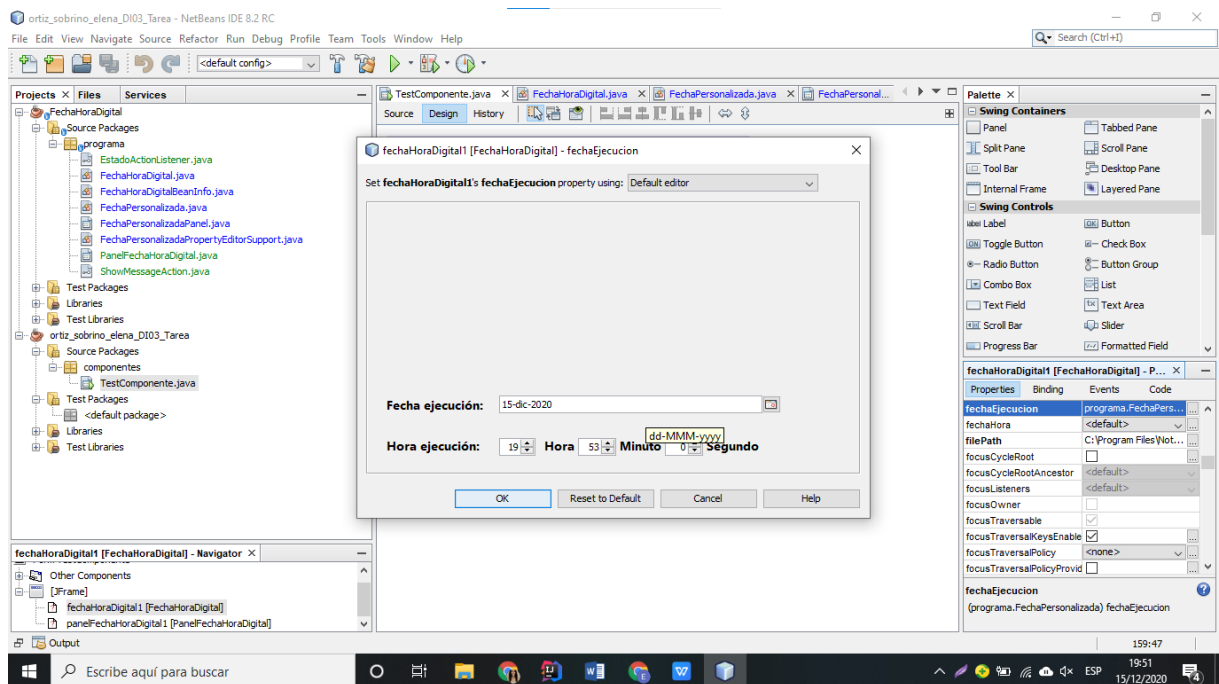
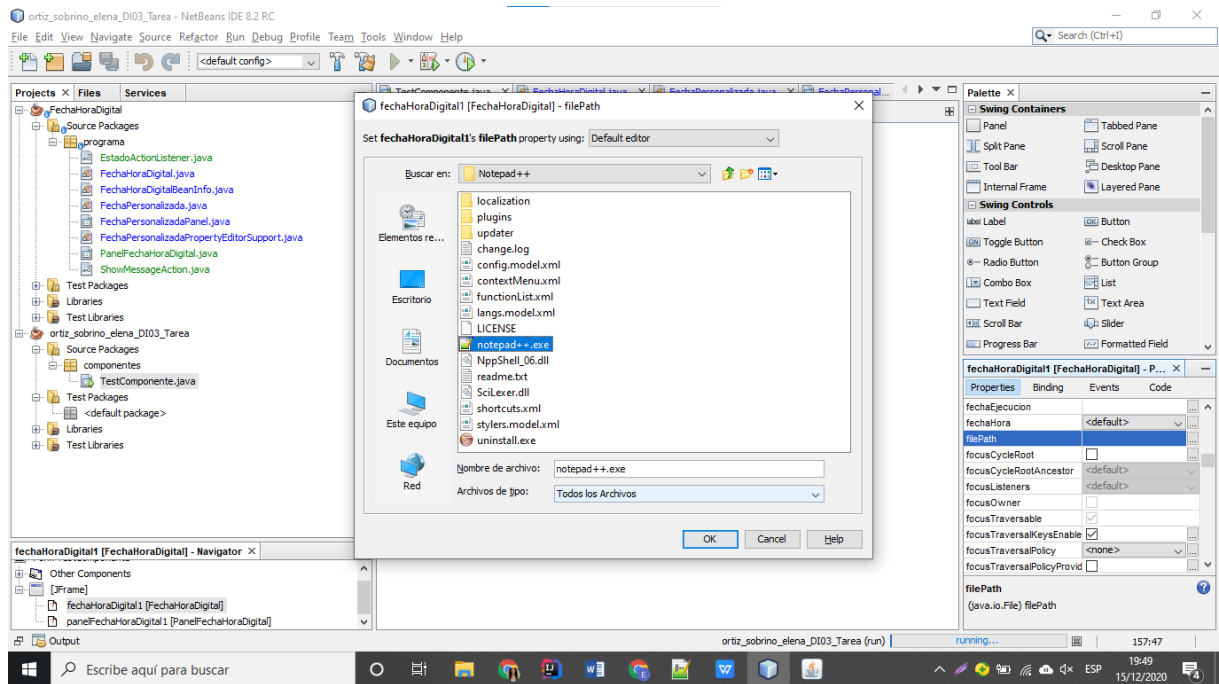


Capturas del programa

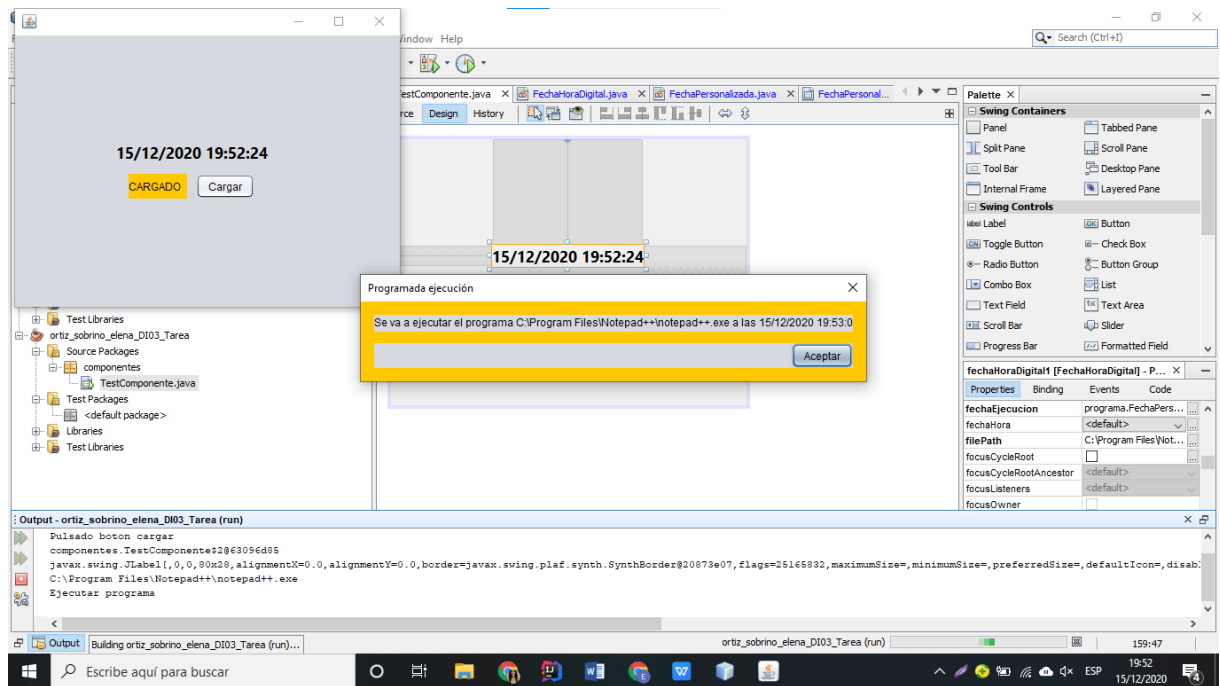
Si no se rellenan los campos filePath o fecha de ejecución cuando pulsas cargar te sale un mensaje de aviso



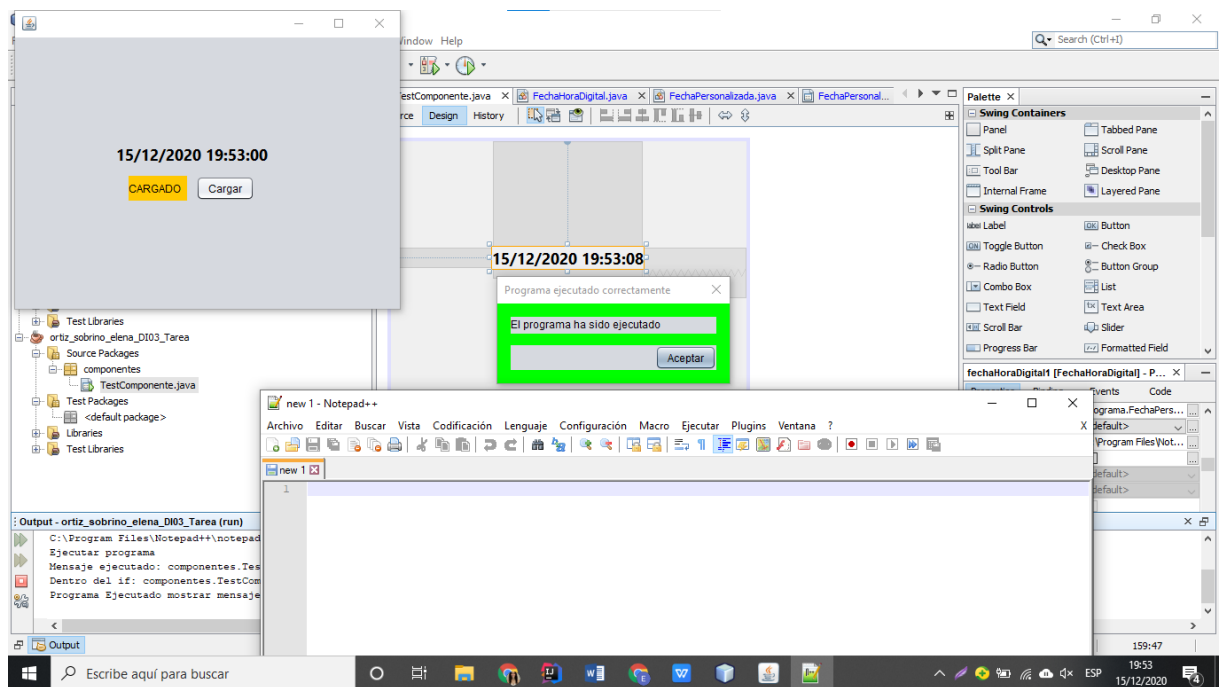
Seleccionamos el programa



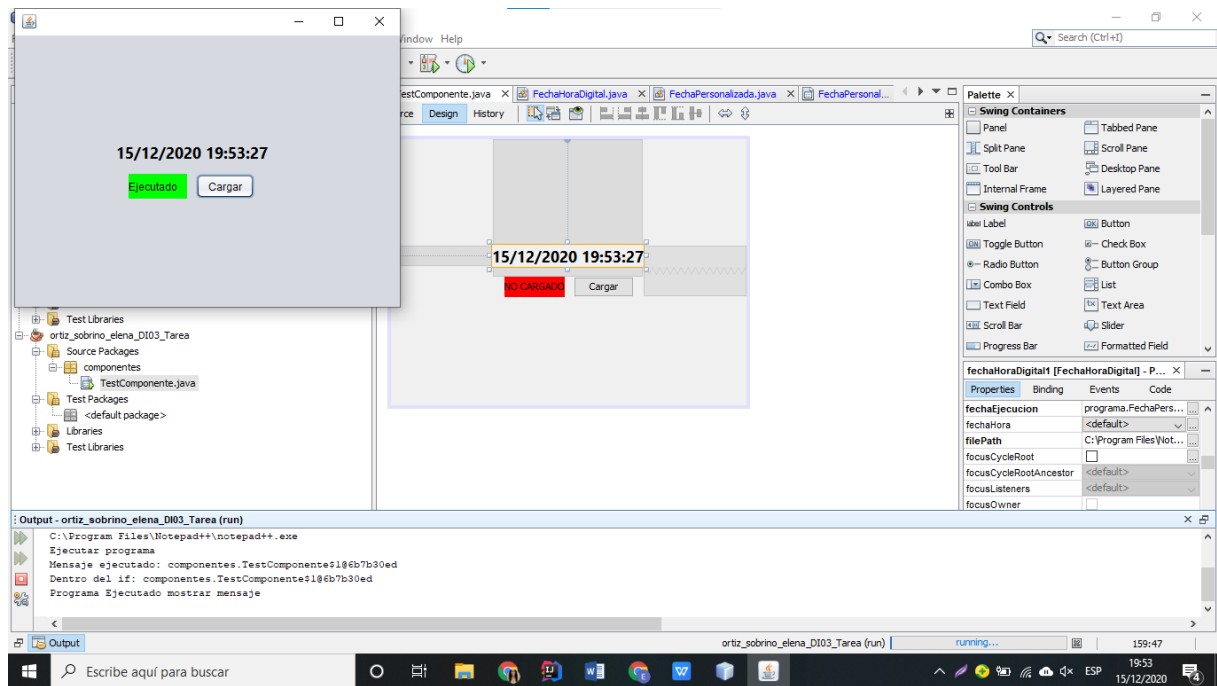
Una vez cargado el programa y puesta la fecha de ejecución y seleccionamos cargar nos sale un mensaje verificando el programa y la hora de ejecución



Cuando llega la hora se ejecuta el programa seleccionado y muestra un mensaje.



Cuando cerramos el mensaje la etiqueta muestra que el programa se ha ejecutado



Compilación del proyecto por consola

