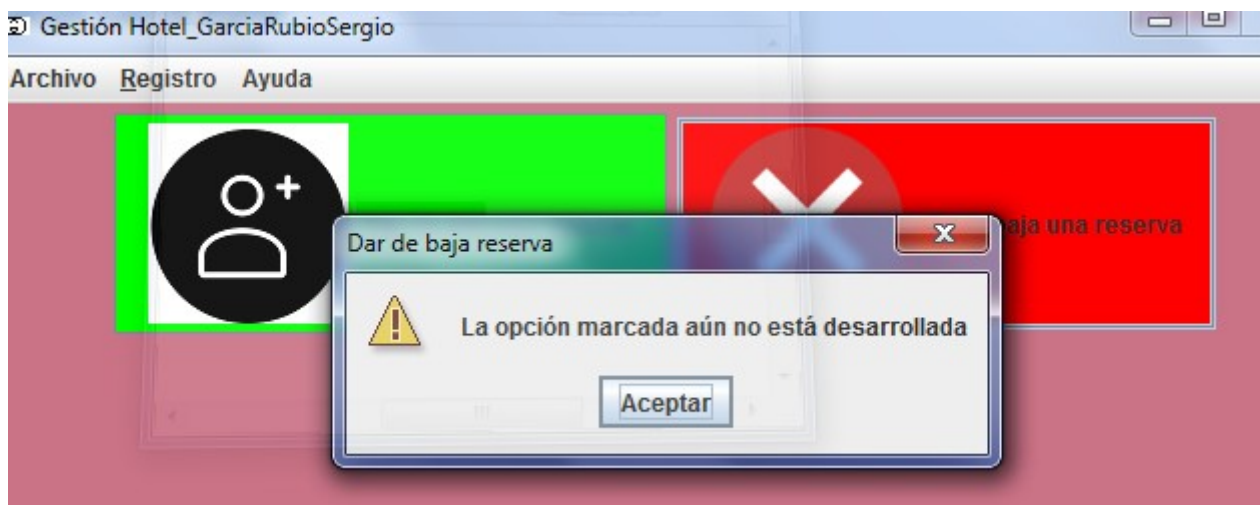
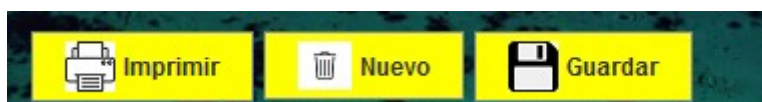


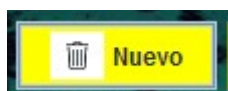
1. Visibilidad: Explico al usuario cuál es el estado del sistema en cada momento, como en este caso que le indico que esa opción no está disponible en este momento



2. Relación con la realidad: Utilizo un lenguaje familiar y apropiado para los usuarios



3. Control y libertad: Ofrezco funciones de rehacer y deshacer que permitan al usuario tener el control de sus interacciones con libertad. Con este botón se puede empezar a rellenar el formulario desde cero.



4. Consistencia y estándares: Establezco una convención lógica, en este caso las opciones del registro alta y baja hacen la misma acción que sus botones homólogos situados en la ventana.



5. Prevención de errores: Ayudo a los usuarios a evitar equivocarse antes de que cometan el error.

En este caso un Tooltip para indicar cómo se tiene que rellenar el campo DNI



6. Reconocimiento: Hago visible todo lo que sea posible, le pongo titulos para que sea facil reconocer los diferentes campos y pongo las instrucciones que deben hacer a la vista con los botones

Hotel Campito del Lago

FORMULARIO RESERVA

NOMBRE: Sergio

APELLIDOS: García Rubio

TELÉFONO:

DNI:

FECHA DE ENTRADA: 2022-11-26

FECHA DE SALIDA: 2022-11-27

Días de estancia: 1

DATOS DE HABITACIÓN

TIPO DE HABITACIÓN: Simple

NÚMERO DE HABITACIONES RESERVADAS: 0

¿NIÑOS?

IMPORTE HABITACION-CLICK AQUÍ PARA VERLO

VER HABITACIÓN 1

VER HABITACIÓN 2

VER HABITACIÓN 3

Imprimir Nuevo Guardar

7. Flexibilidad: Permitoque el sistema pueda adaptarse a los usuarios frecuentes, diseño la realización de tareas avanzadas de manera fluida y eficiente. En este caso he usado atajos par que los usuarios no tengan que hacer click en el boton y puedan accionarlos usando ALT + a , b o r.

```
btnAlta.setMnemonic(KeyEvent.VK_A); // para asociar el boton con la alt + A
btnBaja.setMnemonic(KeyEvent.VK_B); // para asociar el boton con la alt + B
```

8. Estética y minimalismo: Muestro sólo lo necesario y relevante en cada situación para no distraer al usuario con información extra poco relevante. Por ejemplo se puede ocultar el panel de datos imprimidos si se vuelve a pulsar el botón imprimir.

-Imprimir Pulsado:

Hotel Campito del Lago

FORMULARIO RESERVA

NOMBRE: Sergio

APELLIDOS: García Rubio

TELÉFONO:

DNI:

FECHA DE ENTRADA: 2022-11-26

FECHA DE SALIDA: 2022-11-27

Días de estancia: 1

Datos Habitacion
Datos Clientes

Nombre: Sergio
Apellidos: García Rubio
Telefono:
DNI:
Fecha de entrada: 2022-11-26
Fecha de salida: 2022-11-27
Días de estancia: 1

DATOS DE HABITACIÓN

TIPO DE HABITACIÓN: Simple

NÚMERO DE HABITACIONES RESERVADAS: 0

¿NIÑOS?

IMPORTE HABITACION-CLICK AQUÍ PARA VERLO

VER HABITACIÓN 1

VER HABITACIÓN 2

VER HABITACIÓN 3

Imprimir Nuevo Guardar

-Imprimir sin Pulsar:

9. Recuperarse de los errores: Ayudo a los usuarios a reconocer y corregir sus errores, en este caso indico que no tiene relleno los datos obligatorios y por lo tanto no puede guardarlos.

10. Ayuda y documentación: Ofrezco información y ayuda sobre la utilidad del programa mediante ToolTip en los campos que creo que puede haber confusión o desconocimiento. En este ejemplo indico con un ToolTip que se puede ocultar un panel despusando un botón imprimir.

```
// Uso JToggleButton como componente extra que no se usa en la tarea
imprimir = new JToggleButton("Imprimir", imImprimir);
imprimir.setToolTipText(" muestra y esconde el TabbedPane ");
```

-Elección de colores:

Las Gammas de colores que he elegido han sido el contraste entre el **azul** y el **amarillo**. Usando para ello un fondo que potencia y ofrezca una buena visibilidad de los componentes de esos colores.

El amarillo es el propio del sistema y el Azul oscuro es un color personalizado para adaptarse mejor, en concreto es el color con rpg 11,1,97.

-componente libre: JToggleButton imprimir. Tiene el evento de ocultar el panel de datos imprimidos cuando no esta pulsado.