

**Creación de la interfaz
RESERVAS HOTEL**

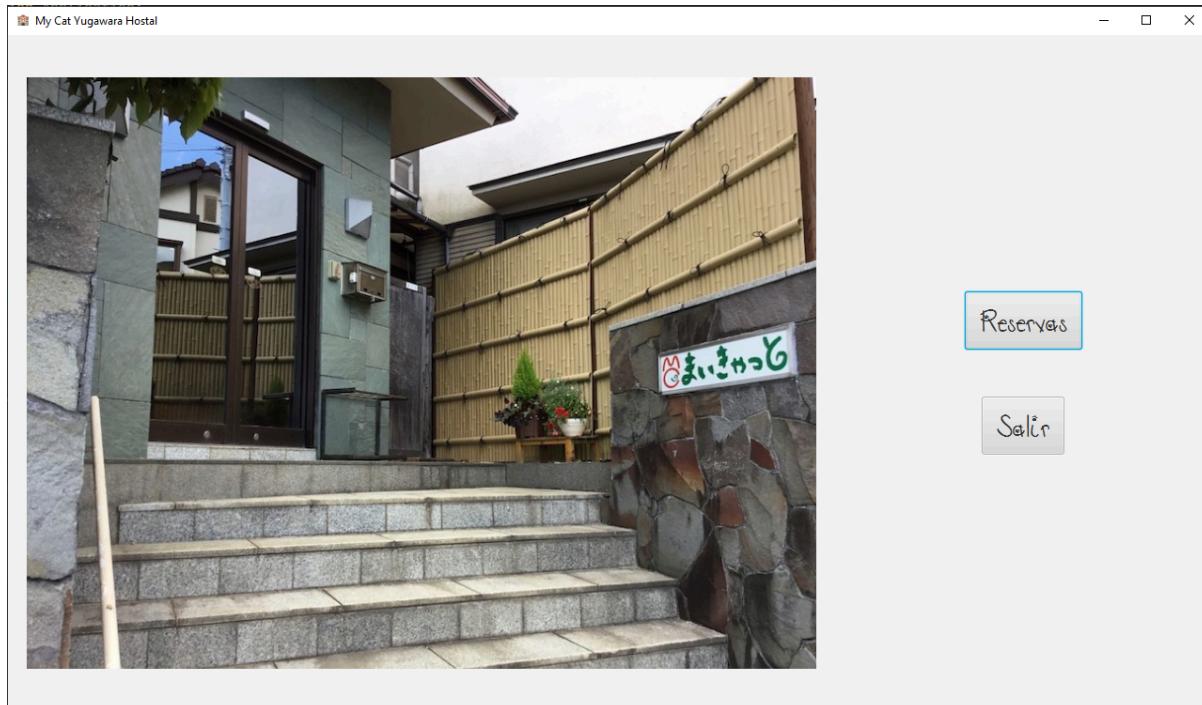
Índice

Introducción	3
1- Creación de la interfaz empleando los componentes más adecuados.	5
2- Cambiar los nombres y ToolTip para los elementos de formulario adecuados	8
3- Distribuir los elementos adecuadamente usando el modo de distribución libre.	16
4- Modificar la fuente de las etiquetas para dar vistosidad a la interfaz.	22
5- Añadir el código para poder abrir el diálogo desde la ventana principal usando el menú y usando un botón a que se gestione el evento.	24
6- Añadir la gestión de eventos para que cuando esté activo el checkbox de fumador se pueda leer el mensaje anteriormente citado y no se muestre si no está activo.	26
7- Añadir las combinaciones de teclas para que el cursor se coloque sobre el campo de texto correspondiente a la combinación (Alt + tecla).	28
Conclusiones	30

Introducción

Este trabajo tiene como objetivo el diseño y desarrollo de una interfaz gráfica de usuario (GUI) para la gestión de reservas en un hotel con temática única, el *My Cat Yugawara*, ubicado en Japón. Este hotel ofrece una experiencia innovadora en la que los huéspedes pueden compartir sus habitaciones con sus propios gatos o con los gatos del hostal, promoviendo la adopción de estos animales. El desafío ha sido crear una interfaz amigable y funcional que permita a los usuarios realizar reservas de manera intuitiva, gestionar sus preferencias y comunicarse con el hotel de forma eficiente. Para ello, se han empleado los componentes adecuados para asegurar una experiencia visual atractiva y una distribución práctica de los elementos en la interfaz, cumpliendo con los requisitos establecidos para este trabajo. Además, se ha integrado la gestión de eventos, accesibilidad con combinaciones de teclas, y una estética visual que refuerza la temática del hotel, brindando una interacción fluida y coherente con el concepto del *My Cat Yugawara*.

Antes que nada, veamos el resultado final:



La interfaz del menú principal.*

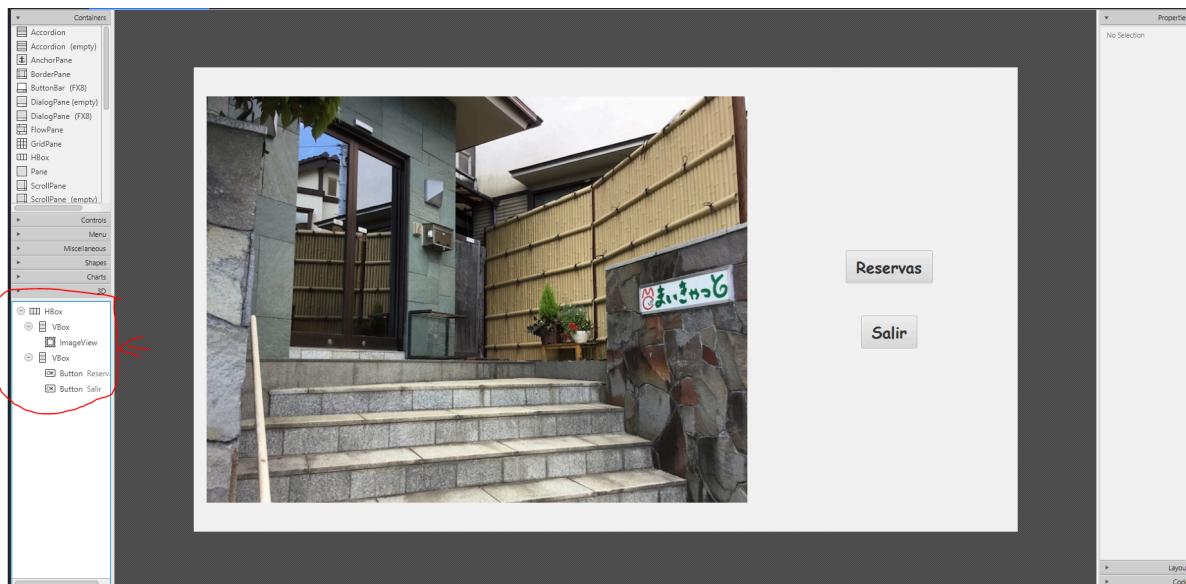
La interfaz de la página de reservas.*

*(Las tipografías son diferentes en la pantalla del scene builder y en el resultado ejecutado dado que se ha usado una tipografía personalizada)

1- Creación de la interfaz empleando los componentes más adecuados.

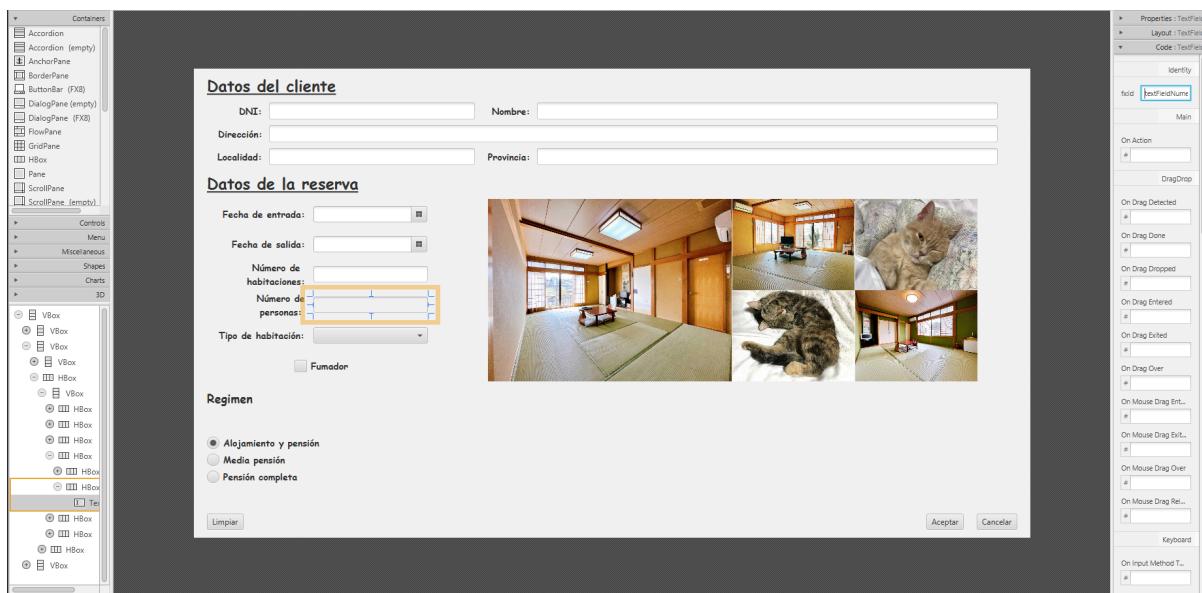
Se han creado dos interfaces, la del menú principal y la de las reservas.

- Menú principal: cuenta con una imagen grande y dos botones. Para ajustarlos correctamente a la página se han utilizado dos contenedores tipo VBOX dentro de un contenedor HBOX para que todo se quede bien ajustado.



En la imagen se puede ver claramente señalada la estructura de la interfaz gráfica.

- Reservas: cuenta con tres bloques principales donde tenemos agrupados los diferentes componentes. En el primero tenemos los datos del cliente, compuesto por una serie de contenedores VBOX y HBOX como en el menú principal para darle forma y estética, y por los componentes, labels y textfields. En el segundo tenemos los datos de la reserva, donde tenemos a la izquierda los componentes del formulario con datepickers(para las fechas de entrada y salida), labels, textfields, un choicebox(para elegir el tipo de habitación), y un checkbox(para indicar si es fumador). En el tercero tenemos el régimen de la estancia, usando un label y tres radiobutton englobados en un toglegroup y tenemos a la derecha dos labels ocultos, uno con el mensaje que aparecerá si se eligen habitaciones para fumadores, y otro para las alertas, que nos dirá cuando se introduzcan valores erróneos o cuando se envíe el formulario. Finalmente tenemos los tres botones, el de limpiar para limpiar todos los campos, el de aceptar para enviar el formulario y el de cancelar para volver a la página del menú principal.



En la imagen se ve la interfaz de las Reservas con sus diversos componentes.



En esta imagen podemos ver la interfaz lanzada y en la que aparecen el mensaje si se eligen habitaciones de fumador y el mensaje de alerta por haber intentado pulsar en el botón de aceptar con los campos vacíos. Hay más restricciones.

```
private boolean validateFields() { 1 usage new *
    String warningMessage = "";
    if (!isValidDNI(textFieldDNI.getText())) {
        warningMessage = "DNI inválido.";
    }
    if (containsNumbers(textFieldNombre.getText())) {
        warningMessage = "El campo Nombre no puede contener números.";
    }
    if (containsNumbers(textFieldLocalidad.getText())) {
        warningMessage = "El campo Localidad no puede contener números.";
    }
    if (containsNumbers(textFieldProvincia.getText())) {
        warningMessage = "El campo Provincia no puede contener números.";
    }
    if (datePickerFechaEntrada.getValue() == null || datePickerFechaSalida.getValue() == null ||
        datePickerFechaSalida.getValue().isBefore(datePickerFechaEntrada.getValue())) {
        warningMessage = "Fechas de entrada y salida inválidas.";
    }
    if (!isNumeric(textFieldNumeroHabitaciones.getText()) || Integer.parseInt(textFieldNumeroHabitaciones.getText()) > 10) {
        warningMessage = "Número de habitaciones inválido (máximo 10).";
    }
    if (textFieldNumeroPersonas.getText().isEmpty() || !isNumeric(textFieldNumeroPersonas.getText())) {
        warningMessage = "El campo Número de personas no puede estar vacío y debe contener solo números.";
    }
    if (textFieldDNI.getText().isEmpty() || textFieldNombre.getText().isEmpty() ||
        textFieldDireccion.getText().isEmpty() || textFieldLocalidad.getText().isEmpty() ||
        textFieldProvincia.getText().isEmpty() || datePickerFechaEntrada.getValue() == null ||
        datePickerFechaSalida.getValue() == null || textFieldNumeroHabitaciones.getText().isEmpty()) {
        warningMessage = "Todos los campos deben estar completos.";
    }
    if (!warningMessage.isEmpty()) {
        labelAviso.setText(warningMessage.toString());
        return false;
    }
    return true;
}
```

Método que gestiona las restricciones de los campos.

2- Cambiar los nombres y ToolTip para los elementos de formulario adecuados

Todos los elementos de la interfaz tienen los nombres cambiados y los campos tienen ToolTips

```
@FXML  
private Label labelDatosReserva;  
  
@FXML  
private Label labelDatosCliente;  
  
@FXML  
private Label labelDNI;  
  
@FXML  
private TextField textFieldDNI;  
  
@FXML  
private Label labelNombre;  
  
@FXML  
private TextField textFieldNombre;  
  
@FXML  
private Label labelDireccion;  
  
@FXML  
private TextField textFieldDireccion;  
  
@FXML  
private Label labelLocalidad;  
  
@FXML  
private TextField textFieldLocalidad;  
  
@FXML  
private Label labelProvincia;  
  
@FXML  
private TextField textFieldProvincia;  
  
@FXML  
private Label labelFechaEntrada;
```

```
@FXML  
private DatePicker datePickerFechaEntrada;  
  
@FXML  
private Label labelFechaSalida;  
  
@FXML  
private DatePicker datePickerFechaSalida;  
  
@FXML  
private Label labelNumeroHabitaciones;  
  
@FXML  
private TextField textFieldNumeroHabitaciones;  
  
@FXML  
private Label labelNumeroPersonas;  
  
@FXML  
private TextField textFieldNumeroPersonas;  
  
@FXML  
private Label labelTipoHabitacion;  
  
@FXML  
private ChoiceBox<String> choiceBoxTipoHabitacion;  
  
@FXML  
private CheckBox checkBoxfumador;  
  
@FXML  
private Label labelRegimen;  
  
@FXML  
private ToggleGroup toggleGroupRegimen;  
  
@FXML  
private RadioButton radioButtonAlojamientoPension;  
  
@FXML  
private RadioButton radioButtonMediaPension;
```

```
@FXML  
private RadioButton radioButtonPensionCompleta;  
@FXML  
private Label labelFumador;  
@FXML  
private Label labelAviso;  
@FXML  
private Button buttonLimpiar;  
@FXML  
private Button buttonAceptar;  
@FXML  
private Button buttonCancelar;
```

En estas imágenes podemos ver la declaración de los diferentes nombres de los elementos de la interfaz de Reservas.

```
@FXML  
private Button botonReservas;  
@FXML  
private Button botonSalir;
```

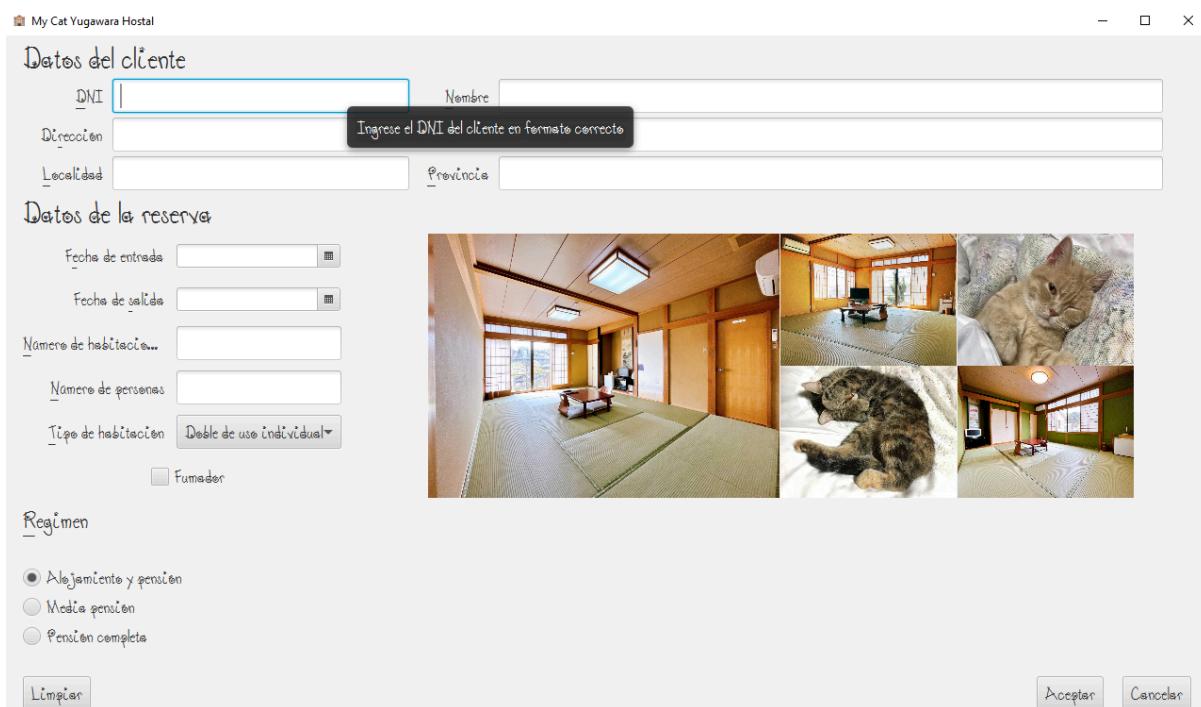
En esta imagen podemos ver los botones declarados de la interfaz del menú principal.

Ambos están separados porque ambos están en sus correspondientes controladores. Estos nombres se asignan en el controlador para poder manejar los distintos eventos que tendremos y en el xml como el id.

```
<VBox alignment="CENTER" prefHeight="200.0" prefWidth="400.0">
  <children>
    <Button fx:id="botonReservas" mnemonicParsing="false" onAction="#onBotonReservas" text="Reservas" textAlignment="CENTER">
      <font>
        <Font name="Comic Sans MS Bold" size="24.0" />
      </font>
      <VBox.margin>
        <Insets bottom="50.0" />
      </VBox.margin>
    </Button>
    <Button fx:id="botonSalir" mnemonicParsing="false" onAction="#onBotonSalir" text="Salir" textAlignment="CENTER">
      <font>
        <Font name="Comic Sans MS Bold" size="24.0" />
      </font>
    </Button>
  </children>
</VBox>
```

En esta imagen podemos ver la declaración en el xml de los nombres de los botones en la interfaz del menú principal. El mismo procedimiento ocurre en la interfaz de Reservas en todos los elementos que han necesitado un nombre para llamarlos desde el controlador.

Por otro lado, se han incluído los ToolTips como se pedía.



My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección
Localidad Ingrese la dirección de residencia del cliente

Datos de la reserva

Fecha de entrada Fecha de salida
Número de habitación... Número de personas...
Tipo de habitación Doble de uso individual Fumador

Regimen

Alojamiento y pensión Media pensión Pensión completa

Limpiar Aceptar Cancelar

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección
Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida Seleccione la fecha de entrada
Número de habitación... Número de personas...
Tipo de habitación Doble de uso individual Fumador

Regimen

Alojamiento y pensión Media pensión Pensión completa

Limpiar Aceptar Cancelar

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección

Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida

Número de habitación...

Número de personas

Tipo de habitación Doble de uso individual
 Fumador



Regímen

Alojamiento y pensión
 Media pensión
 Pensión completa

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección

Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida

Número de habitación...

Número de personas

Tipo de habitación Doble de uso individual
 Fumador



Regímen

Alojamiento y pensión
 Media pensión
 Pensión completa

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección

Localidad Provincia

Datos de la reserva

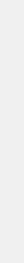
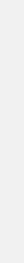
Fecha de entrada Fecha de salida

Número de habitación...

Número de personas

Tipo de habitación

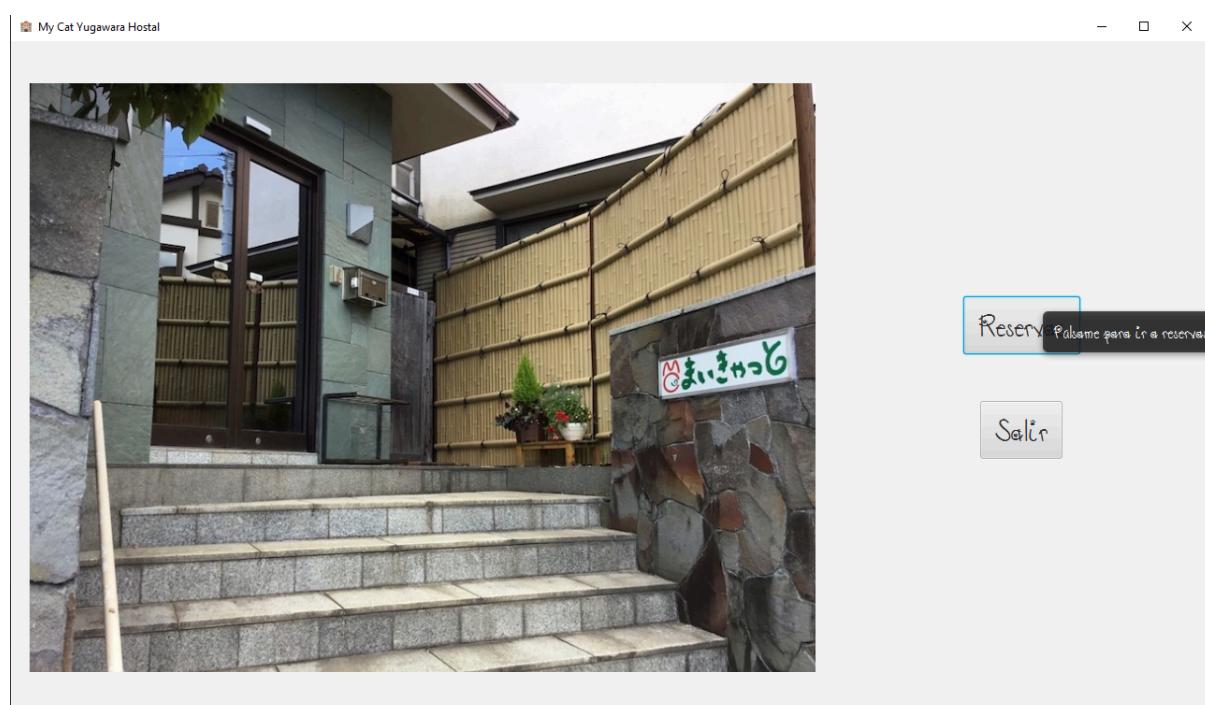
Fumador

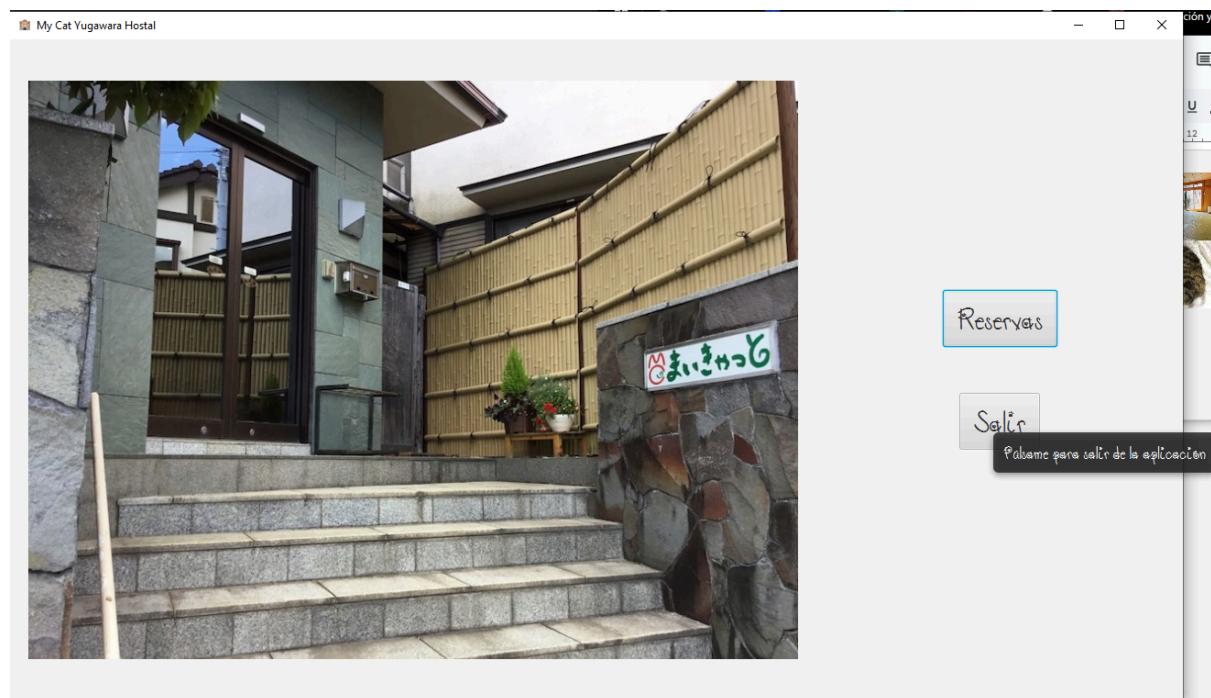


Regímen

Alojamiento y pensión
 Media pensión
 Pensión completa

Confirmar la reserva





En estas imágenes tenemos ejemplos de cómo se ven los ToolTips en nuestra aplicación.

```
public static void setTooltipWithFont(Control control, String tooltipText, Font font) {
    Tooltip tooltip = new Tooltip(tooltipText);
    tooltip.setFont(font);
    control.setTooltip(tooltip);
}
```

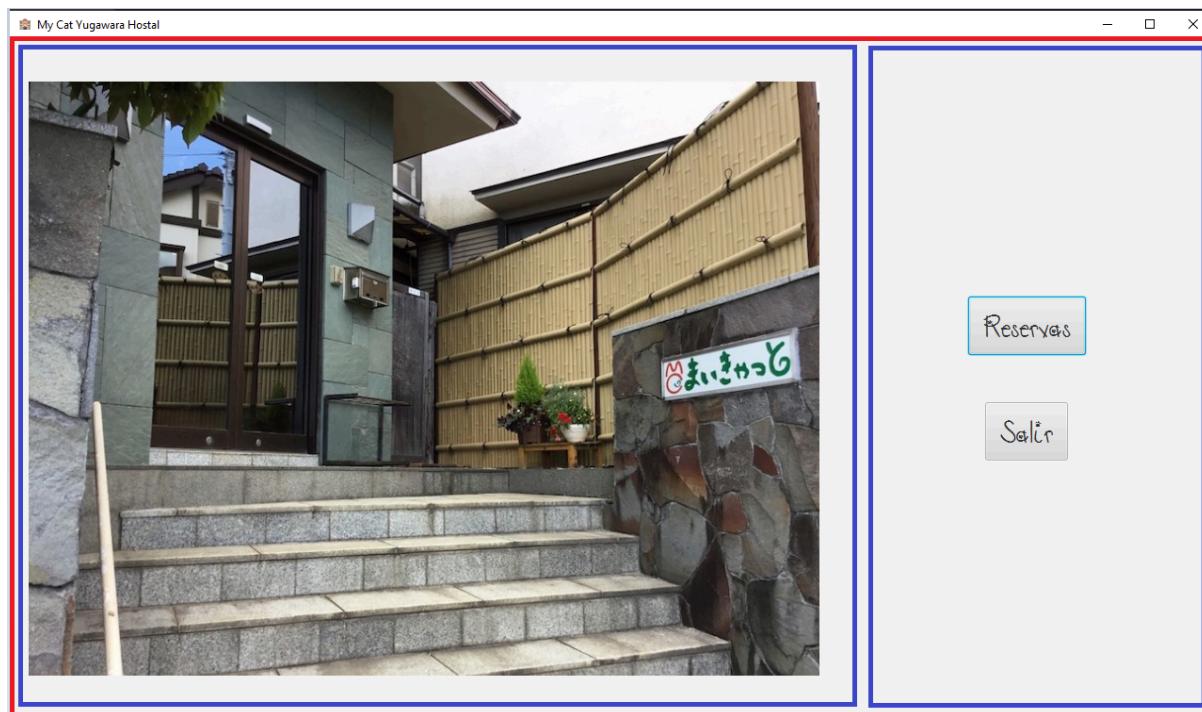
Método al que llamamos para asignar un ToolTip. También asigna la fuente personalizada al ToolTip.

```
AppNavigationService.setTooltipWithFont(textFieldDNI, tooltipText: "Ingrese el DNI del cliente en formato correcto", customFontNormal);
AppNavigationService.setTooltipWithFont(textFieldNombre, tooltipText: "Ingrese el nombre completo del cliente", customFontNormal);
AppNavigationService.setTooltipWithFont(textFieldDireccion, tooltipText: "Ingrese la dirección de residencia del cliente", customFontNormal);
AppNavigationService.setTooltipWithFont(textFieldLocalidad, tooltipText: "Ingrese la localidad de residencia del cliente", customFontNormal);
AppNavigationService.setTooltipWithFont(textFieldProvincia, tooltipText: "Ingrese la provincia de residencia del cliente", customFontNormal);
AppNavigationService.setTooltipWithFont(datePickerFechaEntrada, tooltipText: "Seleccione la fecha de entrada", customFontNormal);
AppNavigationService.setTooltipWithFont(datePickerFechaSalida, tooltipText: "Seleccione la fecha de salida", customFontNormal);
AppNavigationService.setTooltipWithFont(textFieldNumeroHabitaciones, tooltipText: "Ingrese el número de habitaciones (máximo 10)", customFontNormal);
AppNavigationService.setTooltipWithFont(textFieldNumeroPersonas, tooltipText: "Ingrese el número de personas (máximo 4)", customFontNormal);
AppNavigationService.setTooltipWithFont(choiceBoxTipoHabitacion, tooltipText: "Seleccione el tipo de habitación", customFontNormal);
AppNavigationService.setTooltipWithFont(checkBoxFumador, tooltipText: "Marque si el cliente es fumador", customFontNormal);
AppNavigationService.setTooltipWithFont(radioButtonAlojamientoPension, tooltipText: "Seleccione si el régimen es Alojamiento y Desayuno", customFontNormal);
AppNavigationService.setTooltipWithFont(radioButtonMediaPension, tooltipText: "Seleccione si el régimen es Media Pensión", customFontNormal);
AppNavigationService.setTooltipWithFont(radioButtonPensionCompleta, tooltipText: "Seleccione si el régimen es Pensión Completa", customFontNormal);
AppNavigationService.setTooltipWithFont(buttonLimpiar, tooltipText: "Limpia todos los campos", customFontNormal);
AppNavigationService.setTooltipWithFont(buttonAceptar, tooltipText: "Confirma la reserva", customFontNormal);
AppNavigationService.setTooltipWithFont(buttonCancelar, tooltipText: "Cancela y regresa al menú principal", customFontNormal);
```

Aquí tenemos un ejemplo de cómo está aplicado este método en el controlador de la vista de las Reservas.

3- Distribuir los elementos adecuadamente usando el modo de distribución libre.

Para hacer la distribución de los elementos se han usado principalmente contenedores VBOX y HBOX. Se han usado las diferentes alineaciones, como center o center-right.



En la interfaz del menú principal contamos con un HBOX(rojo) principal, que engloba dos VBOX(azul). Ambos VBOX tienen alineación central.

My Cat Yugawara Hostal

Datos del cliente

DNI	<input type="text"/>	Nombre	<input type="text"/>
Dirección	<input type="text"/>		
Localidad	<input type="text"/>	Provincia	<input type="text"/>

Datos de la reserva

Fecha de entrada	<input type="text"/>	
Fecha de salida	<input type="text"/>	
Número de habitación...	<input type="text"/>	
Número de personas	<input type="text"/>	
Tipo de habitación	<input type="button" value="Doble de uso individual"/>	<input type="checkbox"/> Fumador

Regimen

- Alojamiento y pensión
- Media pensión
- Pensión completa

En la interfaz de reservas tenemos un VBOX general(rojo) y tres VBOX secundarios(azul) dentro del VBOX principal.

My Cat Yugawara Hostal

Datos del cliente

DNI	<input type="text"/>	Nombre	<input type="text"/>
Dirección	<input type="text"/>		
Localidad	<input type="text"/>	Provincia	<input type="text"/>

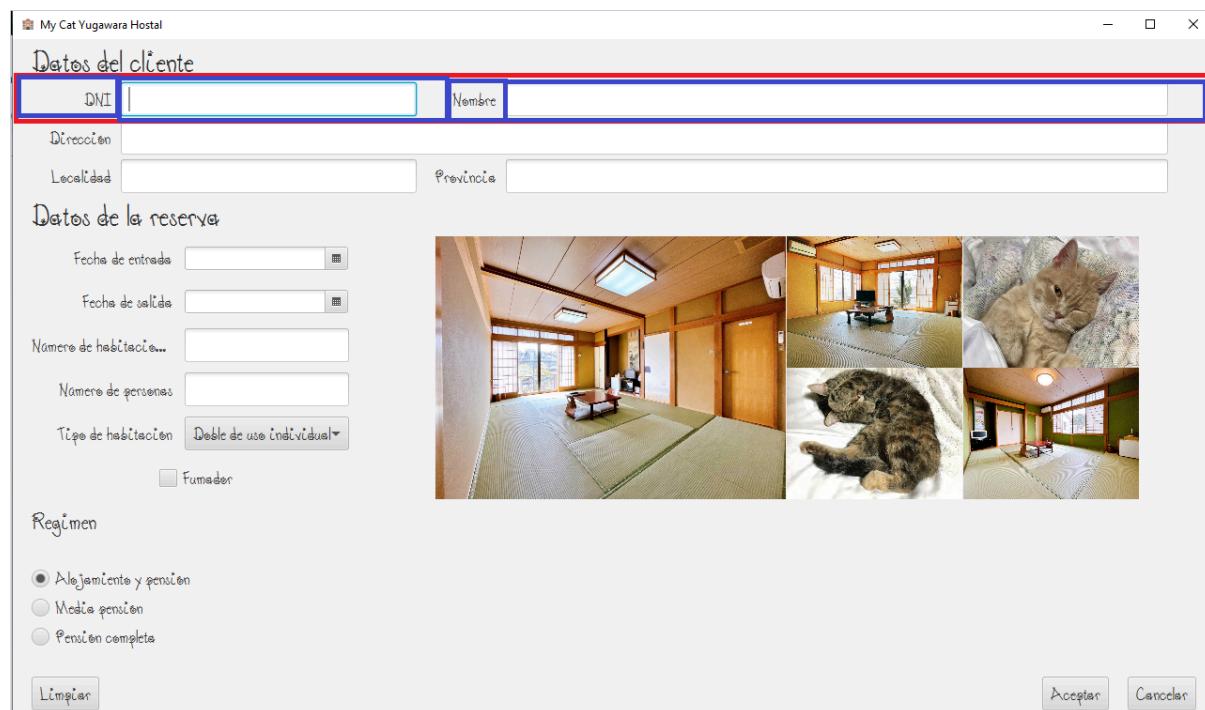
Datos de la reserva

Fecha de entrada	<input type="text"/>	
Fecha de salida	<input type="text"/>	
Número de habitación...	<input type="text"/>	
Número de personas	<input type="text"/>	
Tipo de habitación	<input type="button" value="Doble de uso individual"/>	<input type="checkbox"/> Fumador

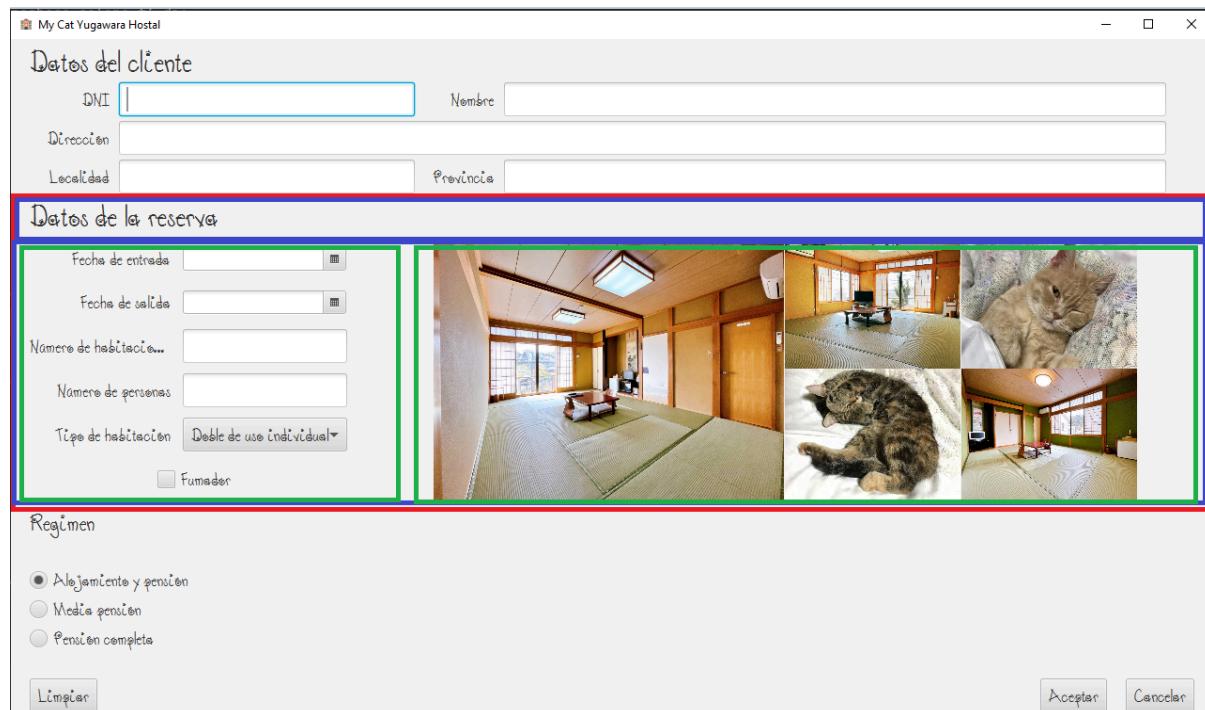
Regimen

- Alojamiento y pensión
- Media pensión
- Pensión completa

En el contenedor secundario superior(rojo) tenemos dos VBOX(azul) y dentro del inferior tenemos tres HBOX(verde) dentro que contienen los label y los textfield.



En los contenedores HBOX(rojo) tenemos una serie de HBOX(azul) independientes, que ayudan a alinear todos los elementos correctamente. Esto se repite en dos demás.



En el siguiente contenedor tenemos un HBOX(rojo) y contiene un VBOX(azul) que es el título y un HBOX(azul) que son el formulario y las imágenes. a su vez, el último HBOX contiene un VBOX(verde) que es el formulario y un HBOX(verde) que son las imágenes.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección

Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida

Número de habitación...

Número de personas

Tipo de habitación

Fumador

Regímen

Alojamiento y pensión
 Media pensión
 Pensión completa

Dentro del VBOX(rojo) del formulario tenemos un HBOX(azul) que contiene otros dos HBOX(verde), que ayudan a alinear correctamente el formulario. Esto se repite con todos los campos salvo el último HBOX(naranja), que no contiene más que el checkbox y que está centrado.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre

Dirección

Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida

Número de habitación...

Número de personas

Tipo de habitación

Fumador

Regímen

Alojamiento y pensión
 Media pensión
 Pensión completa

En el HBOX(rojo) de las imágenes tenemos la imagen central(azul) junto con dos contenedores VBOX(verde) que contienen a su vez otras dos imágenes(naranja) secundarias cada uno.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre
 Dirección
 Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida
 Número de habitación
 Número de personas
 Tipo de habitación Doble de uso individual
 Fumador



Regimen

Alojamiento y pensión
 Media pensión
 Pensión completa

En el último VBOX(rojo) tenemos otros tres HBOX(azul). En el primero solo tenemos un label.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre
 Dirección
 Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida
 Número de habitación
 Número de personas
 Tipo de habitación Doble de uso individual
 Fumador



Regimen

Alojamiento y pensión
 Media pensión
 Pensión completa

En el segundo tenemos el HBOX principal(rojo), dos VOX secundarios(azul), el primero que contiene directamente los 4 radio buttons dentro y el segundo que contiene dos labels(naranja) "ocultos", que solo se muestran si se marca la casilla de fumador o si hay un error o al completar la reserva.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre
Dirección Localidad Provincia

Datos de la reserva

Fecha de entrada Fecha de salida
Número de habitación Número de personas
Tipo de habitación Doble de uso individual Fumador



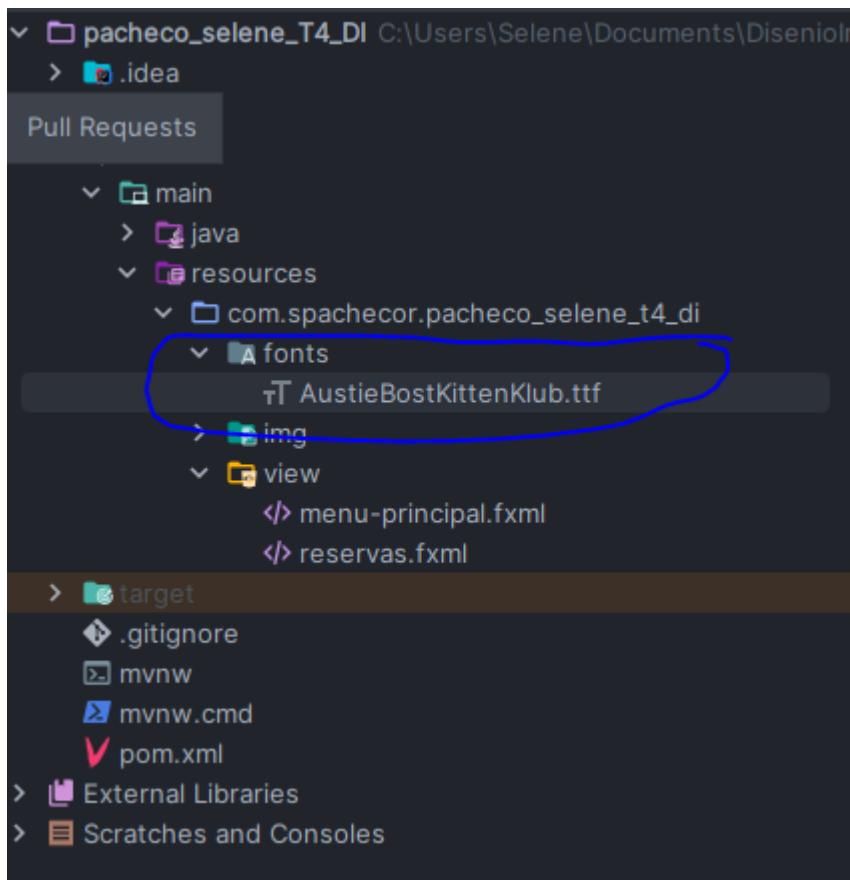
Regímen

Alojamiento y pensión
 Media pensión
 Pensión completa

Por último tenemos el último HBOX(rojo), que contiene a su vez otros dos HBOX(azul) que contienen un botón en el izquierdo y dos en el derecho.

4- Modificar la fuente de las etiquetas para dar vistosidad a la interfaz.

He usado una fuente personalizada llamada [AustieBostKitten](#) que pega con la motivación del hotel. Para ello he seguido los siguientes pasos:



Agregamos en la carpeta de resources una carpeta para fonts y ponemos ahí nuestro archivo .tff.

```
public class Main extends Application { @Seleni
    public static Stage primaryStage; 5 usages
    public static Image icono; 2 usages
    public static Font customFontButton; 4 usages
    public static Font customFontNormal; 23 usages
    public static Font customFontMedium; 2 usages
    @Override @Seleni
    public void start(Stage stage){
        customFontButton = Font.loadFont(Main.class.getResourceAsStream( name: "/com/spachecor/pacheco_selene_t4_di/fonts/AustieBostKittenKlub.ttf"), v: 24);
        customFontNormal = Font.loadFont(Main.class.getResourceAsStream( name: "/com/spachecor/pacheco_selene_t4_di/fonts/AustieBostKittenKlub.ttf"), v: 14);
        customFontMedium = Font.loadFont(Main.class.getResourceAsStream( name: "/com/spachecor/pacheco_selene_t4_di/fonts/AustieBostKittenKlub.ttf"), v: 18);
        primaryStage = stage;
        icono = new Image(Objects.requireNonNull(Main.class.getResourceAsStream( name: "/com/spachecor/pacheco_selene_t4_di/img/icono.png")));
        AppNavigationService.cargarVista(AppNavigationService.menuPrincipalVista);
    }

    public static void main(String[] args) { launch(); }
}
```

En el main se han declarado tres variables estáticas para poderlas usar en todo el programa con las fuentes aplicadas.

```
public static void setFont(Font font, Control... controls) { 5 usages ↗ Selene
    for (Control control : controls) {
        if (control instanceof Labeled) {
            ((Labeled) control).setFont(font);
        } else if (control instanceof TextField) {
            ((TextField) control).setFont(font);
        }
    }
}
```

He creado un método que aplica la fuente al componente que lo requiera en la clase de utilidades.

```
AppNavigationService.setFont(customFontButton, labelDatosReserva, labelDatosCliente);
AppNavigationService.setFont(customFontNormal, labelDNI, textFieldDNI, labelNombre, textFieldNombre,
    labelDireccion, textFieldDireccion, labelLocalidad, textFieldLocalidad,
    labelProvincia, textFieldProvincia, labelFechaEntrada, labelFechaSalida,
    labelNumeroHabitaciones, textFieldNumeroHabitaciones, labelNumeroPersonas, textFieldNumeroPersonas,
    labelTipoHabitacion, checkBoxFumador, radioButtonAlojamientoPension,
    radioButtonMediaPension, radioButtonPensionCompleta, buttonLimpiar,
    buttonAceptar, buttonCancelar);

AppNavigationService.setFont(customFontMedium, labelRegimen, labelFumador, labelAviso);
```

Finalmente, en el controlador de la vista de Reservas llamo al método y aplico la fuente.

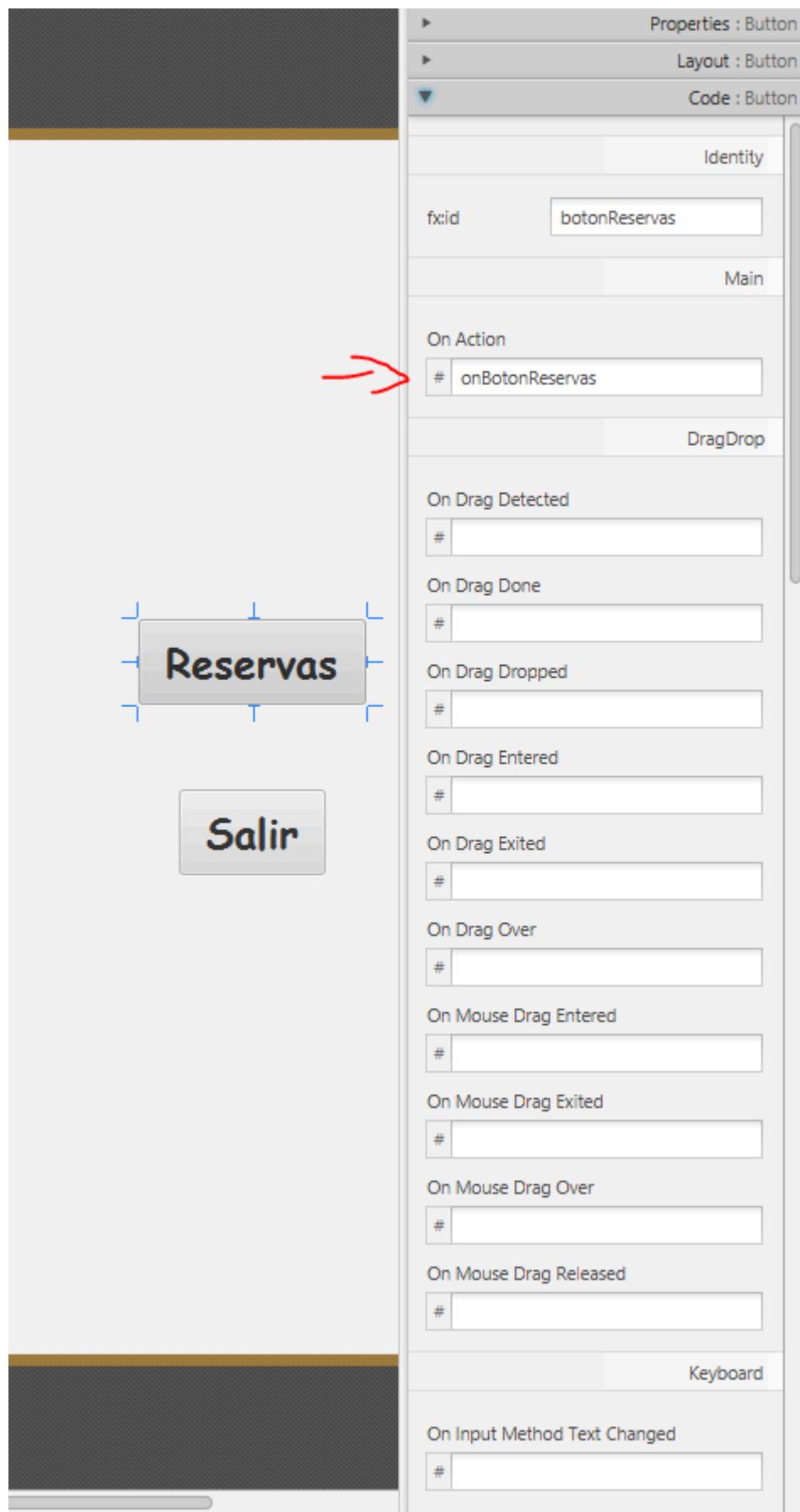
5- Añadir el código para poder abrir el diálogo desde la ventana principal usando el menú y usando un botón a que se gestione el evento.

```
public static boolean cargarVista(String vista){ 3 usages  ↵Selene
try{
    FXMLLoader fxmlLoader = new FXMLLoader(Main.class.getResource( name: "view/"+vista+".fxml"));
    Scene scene = new Scene(fxmlLoader.load(),  v: 1280,  v1: 720);
    primaryStage.setTitle("My Cat Yugawara Hostal");
    primaryStage.getIcons().add(icono);
    primaryStage.setScene(scene);
    primaryStage.show();
    return true;
}catch(IOException e){
    return false;
}
}
```

He creado un método que carga la vista según la cadena que le metamos.

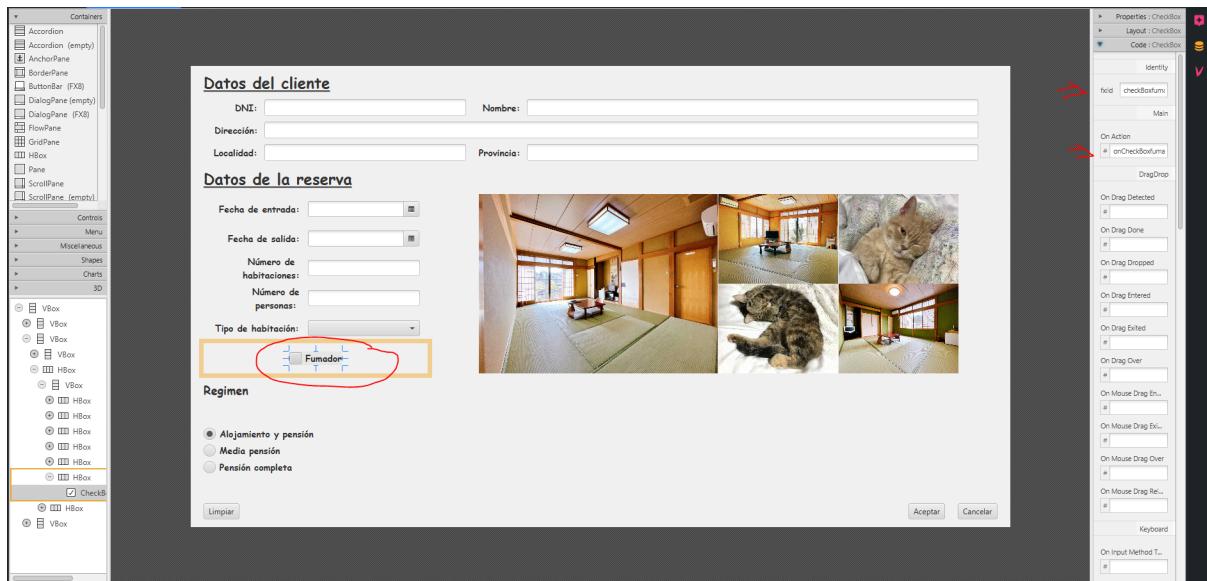
```
@FXML  ↵Selene
protected void onBotonReservas() { AppNavigationService.cargarVista(AppNavigationService.reservasVista); }
```

Llamo al método desde el controlador del Menú principal con el método vinculado al onclick del botón.



Aquí vinculo en el Scene Builder el método con el botón.

6- Añadir la gestión de eventos para que cuando esté activo el checkbox de fumador se pueda leer el mensaje anteriormente citado y no se muestre si no está activo.



Primero comienzo poniendo el nombre al elemento y el evento de onclick, que se activará al pulsar sobre el checkbox.

```
@FXML
protected void onCheckBoxfumador() {
    if (checkBoxFumador.isSelected()) {
        labelFumador.setText("En virtud de la Ley de Sanidad se informa a los clientes de que sólo podrán fumar en las habitaciones\n" +
                            "reservadas para tal fin.");
    } else labelFumador.setText("");
}
```

Defino el método y así quedaría.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre
 Dirección Provincia
 Localidad

Datos de la reserva

Fecha de entrada Fecha de salida
 Número de habitación...
 Número de personas
 Tipo de habitación Doble de uso individual
 Fumador



Alojamiento y pensión
 Media pensión
 Pensión completa

Así se ve sin pulsar.

My Cat Yugawara Hostal

Datos del cliente

DNI Nombre
 Dirección Provincia
 Localidad

Datos de la reserva

Fecha de entrada Fecha de salida
 Número de habitación...
 Número de personas
 Tipo de habitación Doble de uso individual
 Fumador



Alojamiento y pensión
 Media pensión
 Pensión completa

En virtud de la Ley de Sanidad se informa a los clientes de que sólo podrán fumar en las habitaciones reservadas para tal fin.

Así se ve pulsado.

7- Añadir las combinaciones de teclas para que el cursor se coloque sobre el campo de texto correspondiente a la combinación (Alt + tecla).

```
public class ReservasController { @Selene
    public void initialize() {
        AppNavigationService.setTooltipWithFont(datePickerFechaSalida, tooltipText: "Seleccione la fecha de salida", customFontNormal);
        AppNavigationService.setTooltipWithFont(textFieldNumeroHabitaciones, tooltipText: "Ingrese el número de habitaciones (máximo 10)", customFontNormal);
        AppNavigationService.setTooltipWithFont(textFieldNumeroPersonas, tooltipText: "Ingrese el número de personas (máximo 4)", customFontNormal);
        AppNavigationService.setTooltipWithFont(choiceBoxTipoHabitacion, tooltipText: "Seleccione el tipo de habitación", customFontNormal);
        AppNavigationService.setTooltipWithFont(checkBoxFumador, tooltipText: "Marque si el cliente es fumador", customFontNormal);
        AppNavigationService.setTooltipWithFont(radioButtonAlojamientoPension, tooltipText: "Seleccione si el régimen es Alojamiento y Desayuno", customFontNormal);
        AppNavigationService.setTooltipWithFont(radioButtonMediaPension, tooltipText: "Seleccione si el régimen es Media Pensión", customFontNormal);
        AppNavigationService.setTooltipWithFont(radioButtonPensionCompleta, tooltipText: "Seleccione si el régimen es Pensión Completa", customFontNormal);
        AppNavigationService.setTooltipWithFont(buttonLimpiar, tooltipText: "Limpia todos los campos", customFontNormal);
        AppNavigationService.setTooltipWithFont(buttonAceptar, tooltipText: "Confirma la reserva", customFontNormal);
        AppNavigationService.setTooltipWithFont(buttonCancelar, tooltipText: "Cancela y regresa al menú principal", customFontNormal);

        configureLabelWithMnemonic(labelDNI, textWithMnemonic: "_DNI", textFieldDNI); // Alt+D para DNI
        configureLabelWithMnemonic(labelNombre, textWithMnemonic: "_Nombre", textFieldNombre); // Alt+N para Nombre
        configureLabelWithMnemonic(labelDireccion, textWithMnemonic: "_Dir_rección", textFieldDireccion); // Alt+R para Dirección
        configureLabelWithMnemonic(labelLocalidad, textWithMnemonic: "_Localidad", textFieldLocalidad); // Alt+L para Localidad
        configureLabelWithMnemonic(labelProvincia, textWithMnemonic: "_Provincia", textFieldProvincia); // Alt+P para Provincia
        configureLabelWithMnemonic(labelFechaEntrada, textWithMnemonic: "F_echa de entrada", datePickerFechaEntrada); // Alt+E para Fecha de entrada
        configureLabelWithMnemonic(labelFechaSalida, textWithMnemonic: "F_echa de _salida", datePickerFechaSalida); // Alt+S para Fecha de salida
        configureLabelWithMnemonic(labelNumeroHabitaciones, textWithMnemonic: "Número de habitaciones", textFieldNumeroHabitaciones); // Alt+H para Número de habitaciones
        configureLabelWithMnemonic(labelNumeroPersonas, textWithMnemonic: "Número de personas", textFieldNumeroPersonas); // Alt+U para Número de personas
        configureLabelWithMnemonic(labelTipoHabitacion, textWithMnemonic: "Típo de habitacion", choiceBoxTipohabitacion); // Alt+T para Tipo de habitación
        configureLabelWithMnemonic(labelRegimen, textWithMnemonic: "_Regimen", radioButtonAlojamientoPension); // Alt+G para Regimen
    }
}
```

En el método initialize() se llama al método configureLabelWithMnemonic(), que es un método creado por mi.

```
private void configureLabelWithMnemonic(Label label, String textWithMnemonic, Control control) { 11 usages @S
    label.setText(textWithMnemonic);
    label.setMnemonicParsing(true);
    label.setLabelFor(control);

    if (control instanceof TextField) {
        label.setOnKeyPressed(event → control.requestFocus());
    } else if (control instanceof DatePicker) {
        label.setOnKeyPressed(event → control.requestFocus());
    } else if (control instanceof CheckBox) {
        label.setOnKeyPressed(event → control.requestFocus());
    } else if (control instanceof ChoiceBox) {
        label.setOnKeyPressed(event → control.requestFocus());
    } else if (control instanceof RadioButton) {
        label.setOnKeyPressed(event → control.requestFocus());
    }
}
```

El método configureLabelWithMnemonic() configura una etiqueta en JavaFX con un "mnemonic" y la asocia a un control para que, al activar la tecla de acceso rápido, el control gane el foco.

En el bloque condicional, se verifica el tipo del control (TextField, DatePicker, CheckBox, ChoiceBox, RadioButton) y se asigna un manejador de eventos KeyPressed que llama a control.requestFocus() para que el control correspondiente reciba el foco al presionar el "mnemonic". Esto asegura que distintos tipos de controles respondan correctamente al uso de teclas de acceso rápido.

Conclusiones

A lo largo de este trabajo se ha logrado crear una interfaz de usuario adecuada para el contexto del hotel *My Cat Yugawara*. Se ha empleado una distribución libre de los elementos, asegurando que todos los componentes sean fácilmente accesibles y visualmente atractivos para el usuario. Se han incorporado detalles como la personalización de los nombres y ToolTips, la modificación de fuentes para dar vistosidad, y la integración de funcionalidades clave como la apertura de diálogos desde la ventana principal mediante un botón y un menú. Además, se ha incluido la gestión de eventos, como el control del comportamiento del checkbox para los fumadores y la posibilidad de asociar combinaciones de teclas para mejorar la accesibilidad. Este enfoque no solo ha permitido cumplir con los requisitos técnicos del proyecto, sino también ofrecer una interfaz que refleje la filosofía del hotel y proporcione una experiencia de usuario agradable, fomentando la interacción y la adopción de los gatos del hostal.