

# Documentación de Pruebas - TATTOOAGE

## Información del Proyecto

- **Aplicación:** TATTOOAGE - Sistema de gestión de tatuajes
- **Tecnología:** JavaFX
- **Fecha:** Enero 2026

## Índice de Fallos Corregidos

#	Fallo	Tipo de Prueba
1	Error BBDD - App no arranca sin conexión	SISTEMA (Robustez)
2	Ventanas NO modales	SISTEMA (Funcional)
3	Alerts sin icono de aplicación	DISEÑO/USABILIDAD
4	Mnemonic Parsing no funcional	USABILIDAD
5	Botones sin iconos	DISEÑO/USABILIDAD
6	Falta de Tooltips	USABILIDAD
7	Botón VER imagen desplazado	DISEÑO/USABILIDAD

## FALLO 1: Error de Conexión a Base de Datos

### Descripción del Problema

La aplicación no arrancaba cuando no había conexión a la base de datos, lanzando una excepción y cerrándose abruptamente.

### Tipo de Prueba

#### PRUEBA DE SISTEMA - Robustez

Las pruebas de robustez verifican que la aplicación maneje correctamente situaciones de error sin fallar catastróficamente.

### Solución Implementada

1. Se añadieron etiquetas de error en cada panel de tabla
2. Se implementó manejo de excepciones (try-catch) en los métodos de carga
3. Se añadieron flags para controlar el estado de conexión

### Archivos Modificados

`ventanaPrincipal.fxml`

Se añadieron Labels de error para cada tabla:

```
<Label fx:id="lblErrorClientes"
        alignment="CENTER"
        prefHeight="400.0"
        prefWidth="925.0"
        style="-fx-font-size: 16px; -fx-text-fill: #cc0000; -fx-background-color:
#ffe6e6; -fx-padding: 20;"
        text="Error de conexion: No se pudo conectar con la base de datos."
        visible="false"
        managed="false"
        wrapText="true"
        VBox.vgrow="ALWAYS" />
```

**Nota:** El atributo `managed="false"` es crucial para que el Label no ocupe espacio en el layout cuando está oculto.

### miControlador.java

Se añadieron variables de control y manejo de excepciones:

```
// Variables de control de errores
@FXML private Label lblErrorClientes;
@FXML private Label lblErrorTatuadores;
@FXML private Label lblErrorCitas;

private boolean errorConexionClientes = false;
private boolean errorConexionTatuadores = false;
private boolean errorConexionCitas = false;

// Método cargarClientes() con manejo de errores
private void cargarClientes() {
    listaClientes.clear();
    try {
        listaClientes.addAll(clienteDAO.cargarClientes());
        errorConexionClientes = false;
        lblErrorClientes.setVisible(false);
        lblErrorClientes.setManaged(false);
        tableViewClientes.setVisible(true);
        tableViewClientes.setManaged(true);
    } catch (Exception e) {
        errorConexionClientes = true;
        lblErrorClientes.setVisible(true);
        lblErrorClientes.setManaged(true);
        tableViewClientes.setVisible(false);
        tableViewClientes.setManaged(false);
    }
}
```

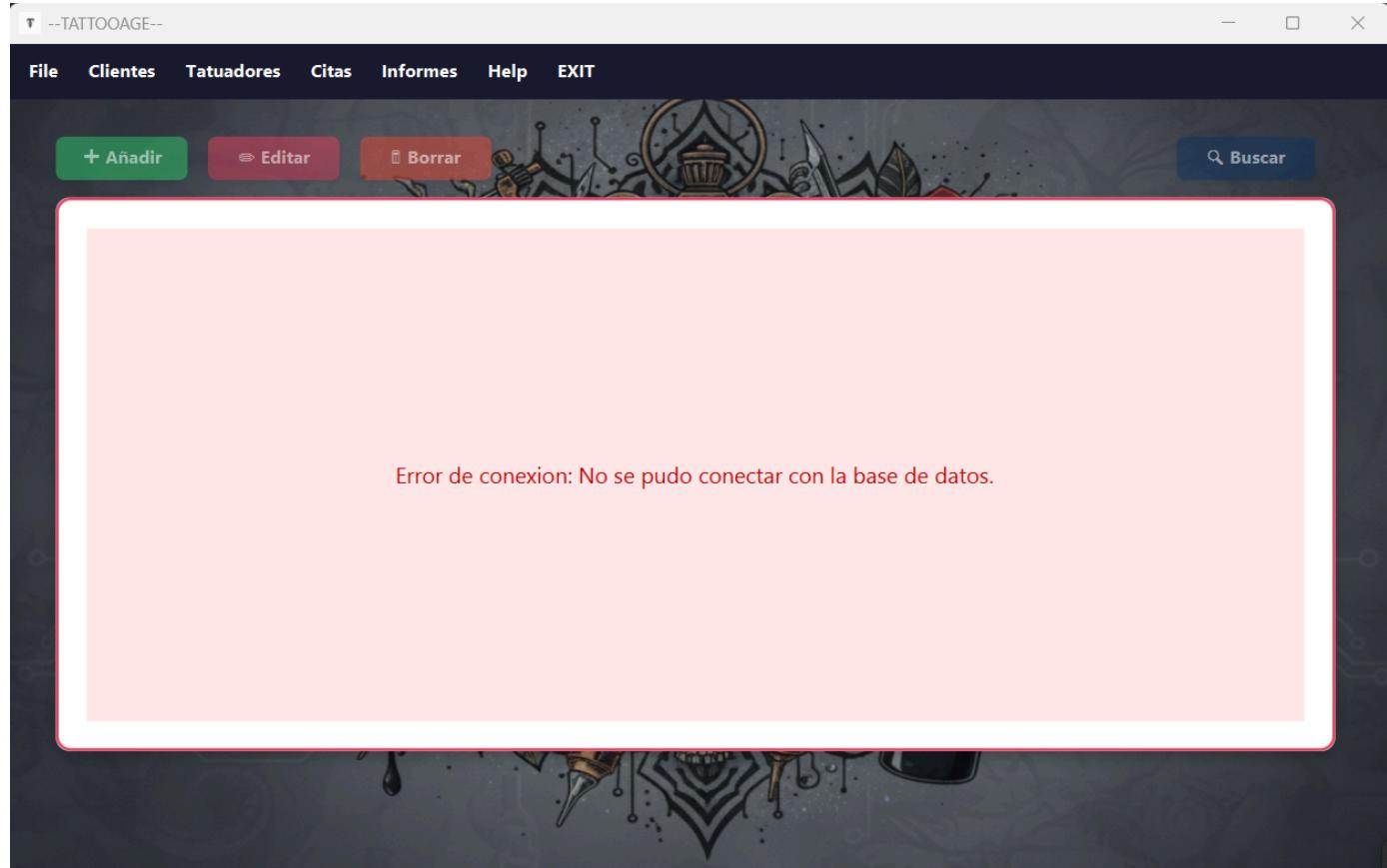
## Capturas de Pantalla

- **Antes:** La aplicación se cerraba con error

```
c:\Users\User\Desktop\FPDAM\SEGUNDO CURSO\INTERFACES\Tema2\ proyecto_app_primerTrimestre>
c:\Users\User\Desktop\FPDAM\SEGUNDO CURSO\INTERFACES\Tema2\ proyecto_app_primerTrimestre> c: && cd "c:\Users\User\Desktop\FPDAM\SEGUNDO CURSO\INTERFACES\Tema2\ proyecto_app_primerTrimestre" && cmd /C "<C:\Program Files\java\jdk-24\bin\java.exe" -agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:60227 @C:\Users\User\AppData\Local\Temp\cp_800yizdrjkxtbn3birvxs5wd.argfile com.javaFx.Main "
WARNING: package com.sun.javafx.sg.prism.web not in javafx.graphics
WARNING: A native method in com.sun.webkit.WebPage has been bound
WARNING: com.sun.webkit.WebPage::twkInitWebCore is declared in module javafx.web (file:/C:/javafx-sdk-25/lib/javafx.web.jar)
WARNING: Use --enable-native-access=javafx.web to avoid a warning for native methods declared in this module
WARNING: Restricted methods will be blocked in a future release unless native access is enabled

No se conectó con la base de datos error: Communications link failure
```

- **Después:** Se muestra mensaje de error en el panel correspondiente



---

## FALLO 2: Ventanas NO Modales

### Descripción del Problema

Las ventanas secundarias (añadir, editar, buscar, etc.) no eran modales, permitiendo al usuario abrir múltiples instancias y seguir interactuando con la ventana principal.

### Tipo de Prueba

#### PRUEBA DE SISTEMA – Funcional

Las pruebas funcionales verifican que las características de la aplicación funcionan según lo especificado.

### Solución Implementada

Se añadió `stage.initModality(Modality.APPLICATION_MODAL)` a todos los métodos que abren ventanas secundarias.

## Archivos Modificados

### miControlador.java

Se modificaron los siguientes métodos:

```
// Método abrirVentana()
private void abrirVentana(String fxml, String titulo) {
    try {
        FXMLLoader loader = new FXMLLoader(getClass().getResource("/" + fxml));
        Parent root = loader.load();
        Stage stage = new Stage();
        stage.setTitle(titulo);
        stage.setScene(new Scene(root));
        stage.initModality(Modality.APPLICATION_MODAL); // AÑADIDO
        StageUtils.setAppIcon(stage);
        stage.showAndWait();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

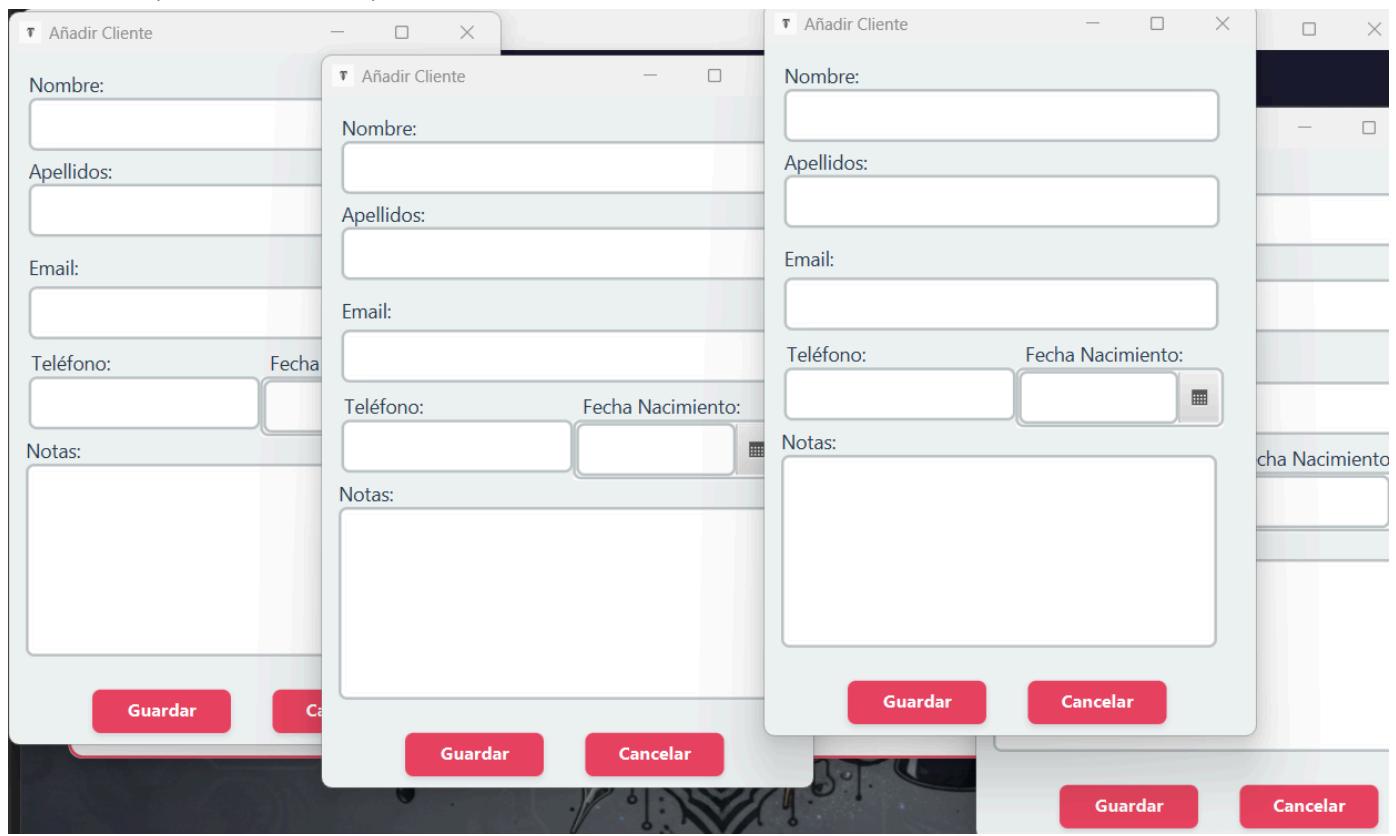
// Método abrirVentanaBusqueda()
private void abrirVentanaBusqueda(String fxml, String titulo) {
    // ... código similar con initModality añadido
    stage.initModality(Modality.APPLICATION_MODAL); // AÑADIDO
}

// Método menuExit()
public void menuExit() {
    // ... código similar con initModality añadido
    stage.initModality(Modality.APPLICATION_MODAL); // AÑADIDO
}

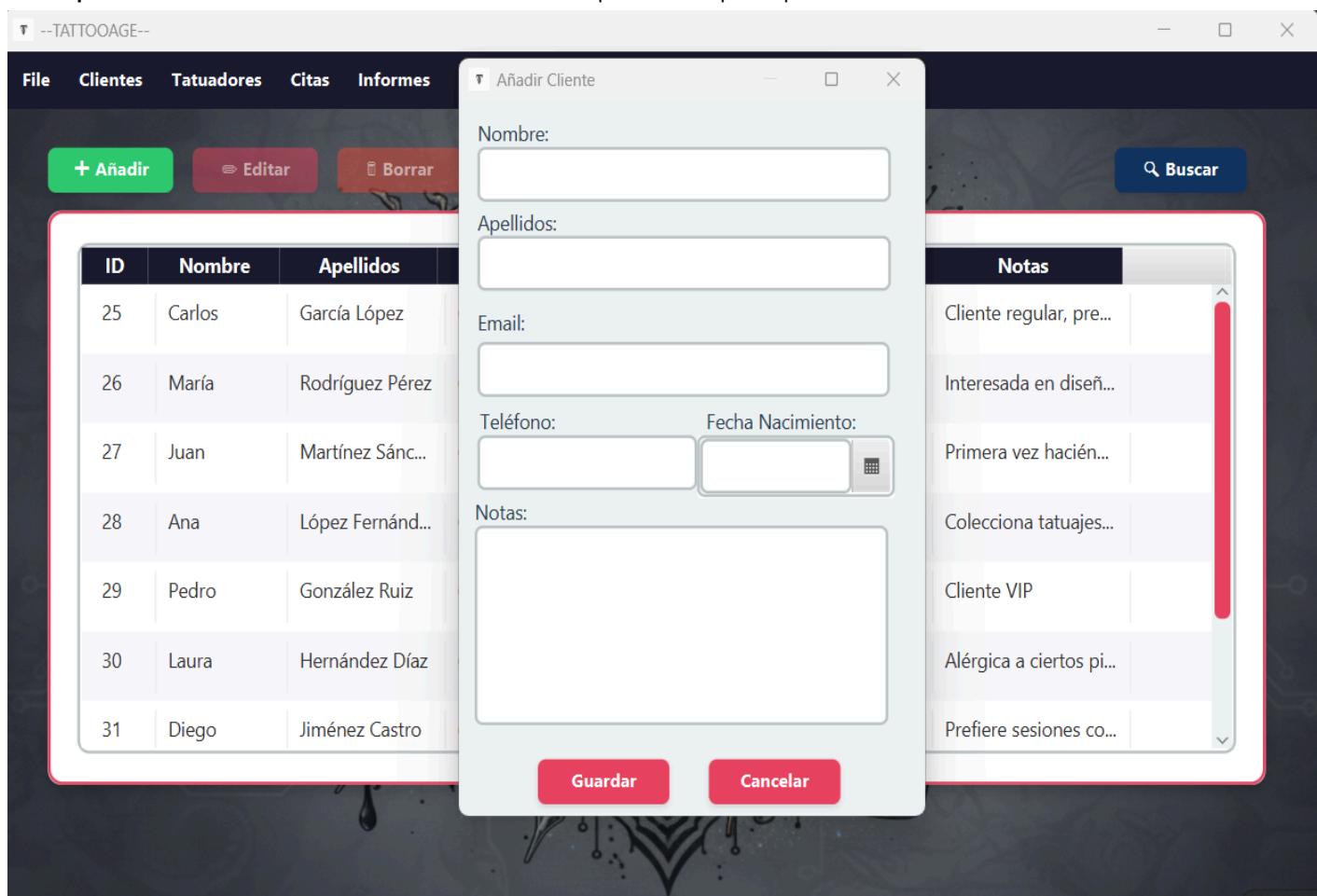
// Método menuGuia()
public void menuGuia() {
    // ... código similar con initModality añadido
    stage.initModality(Modality.APPLICATION_MODAL); // AÑADIDO
}
```

## Capturas de Pantalla

- **Antes:** Se podían abrir múltiples ventanas



- **Después:** Solo una ventana secundaria activa, bloqueando la principal



---

## FALLO 3: Alerts sin Icono de Aplicación

### Descripción del Problema

Los cuadros de diálogo Alert no mostraban el icono de la aplicación en la barra de título.

### Tipo de Prueba

#### PRUEBA DE DISEÑO/USABILIDAD

Las pruebas de usabilidad verifican que la interfaz sea coherente y profesional.

### Solución Implementada

Se añadió el icono de la aplicación a todos los Alerts utilizando `StageUtils.setAppIcon()`.

### Archivos Modificados

#### Todos los controladores con Alerts:

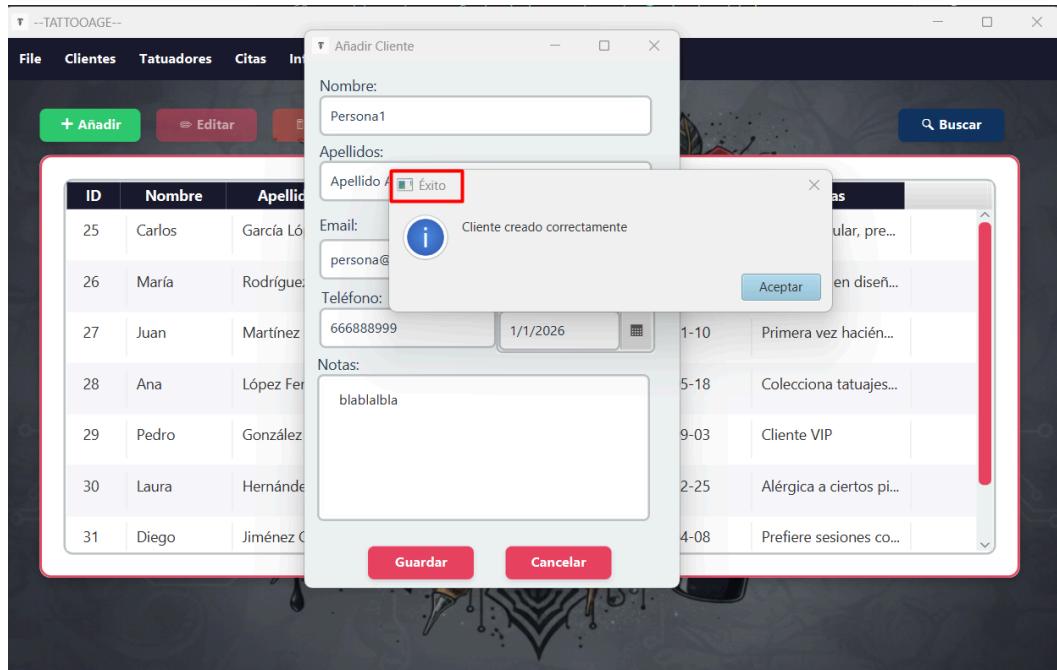
- ventanaAEClienteController.java
- ventanaAETatuadorController.java
- ventanaAECitaController.java
- ventanaBClienteController.java
- ventanaBTatuadorController.java
- ventanaBCitaController.java
- ErrorUtils.java
- miControlador.java

#### Código añadido a cada Alert:

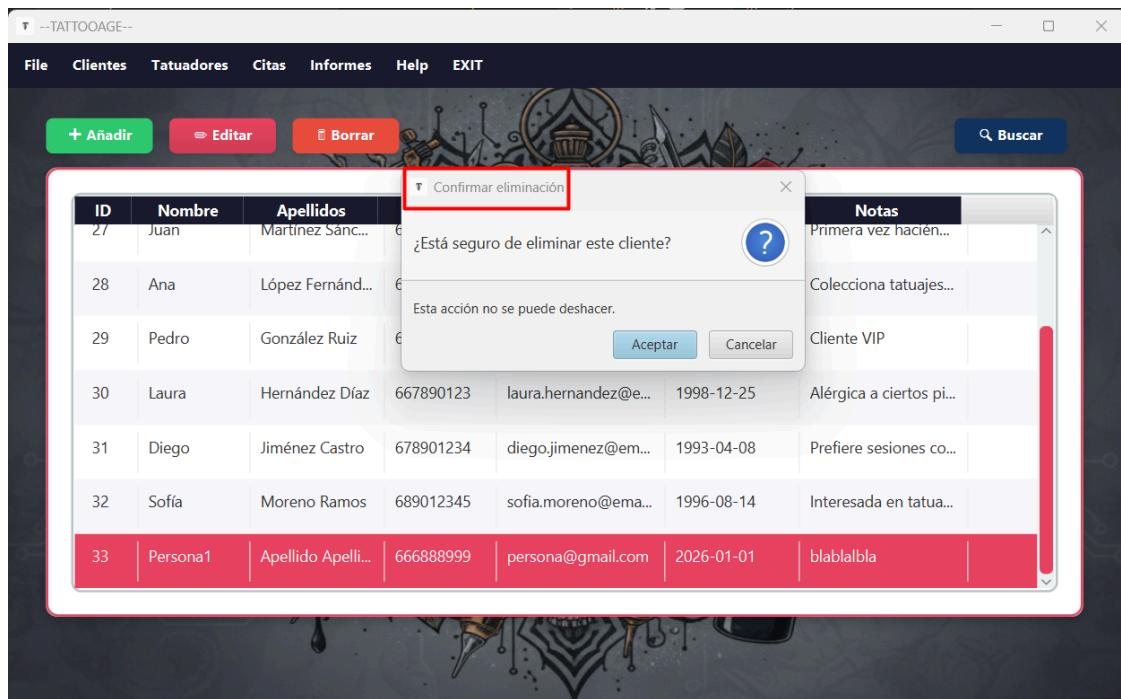
```
private void mostrarAlerta(String titulo, String mensaje, Alert.AlertType tipo) {  
    Alert alerta = new Alert(tipo);  
    alerta.setTitle(titulo);  
    alerta.setHeaderText(null);  
    alerta.setContentText(mensaje);  
  
    // CÓDIGO AÑADIDO - Icono de aplicación  
    Stage alertStage = (Stage) alerta.getDialogPane().getScene().getWindow();  
    StageUtils.setAppIcon(alertStage);  
  
    alerta.showAndWait();  
}
```

## Capturas de Pantalla

- Antes: Alert con icono genérico de Java



- Después: Alert con icono de TATTOOAGE



## FALLO 4: Mnemonic Parsing No Funcional

### Descripción del Problema

Los botones tenían `mnemonicParsing="true"` pero no funcionaban los atajos de teclado porque faltaba el underscore (\_) antes de la letra del atajo.

### Tipo de Prueba

#### PRUEBA DE USABILIDAD

Las pruebas de usabilidad verifican que los atajos de teclado funcionen correctamente para mejorar la accesibilidad.

### Solución Implementada

1. Se añadieron underscores (\_) a los textos de los botones para activar mnemonics (Alt+letra)
2. Se añadió `defaultButton="true"` para activar con Enter
3. Se añadió `cancelButton="true"` para activar con Escape

### Archivos Modificados

#### ventanaPrincipal.fxml - Menús:

```
<Menu mnemonicParsing="true" text="_File">
<Menu mnemonicParsing="true" text="_Clientes">
<Menu mnemonicParsing="true" text="_Tatuadores">
<Menu mnemonicParsing="true" text="C_itas">
<Menu mnemonicParsing="true" text="I_nformes">
```

```
<Menu mnemonicParsing="true" text="_Help">
<Menu mnemonicParsing="true" text="_EXIT">
```

### ventanaPrincipal.fxml - Botones principales:

```
<Button text="⊕ _Añadir" />      <!-- Alt+A -->
<Button text="-pencil _Editar" />    <!-- Alt+E -->
<Button text="trash _Borrar" />     <!-- Alt+B -->
<Button text="magnifying-glass _Buscar" /> <!-- Alt+U -->
<Button text="refresh _Resetear filtro" /> <!-- Alt+R -->
```

### Ventanas secundarias (ejemplo ventanaAECliente.fxml):

```
<!-- ANTES -->
<Button text="Guardar" />
<Button text="Cancelar" />

<!-- DESPUÉS -->
<Button text="diskette _Guardar" defaultButton="true" /> <!-- Enter o Alt+G -->
<Button text="cross _Cancelar" cancelButton="true" /> <!-- Escape o Alt+C -->
```

### ventanaExit.fxml:

```
<Button text="checkmark _Sí" defaultButton="true" /> <!-- Enter o Alt+S -->
<Button text="cross _No" cancelButton="true" /> <!-- Escape o Alt+N -->
```

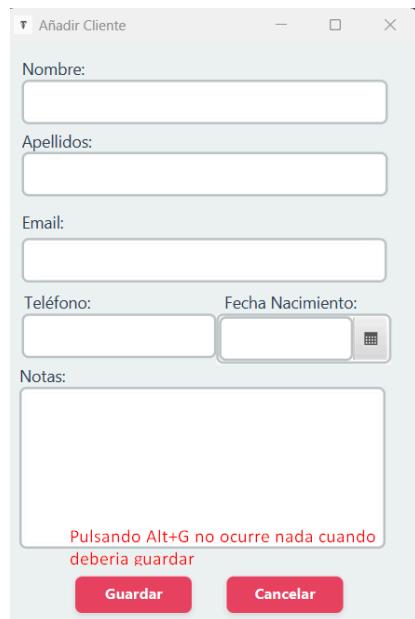
### Tabla de Atajos Implementados

Ventana	Botón	Atajo Alt	Tecla especial
Principal	Añadir	Alt+A	-
Principal	Editar	Alt+E	-
Principal	Borrar	Alt+B	-
Principal	Buscar	Alt+U	-
Principal	Resetear	Alt+R	-
Añadir/Editar	Guardar	Alt+G	Enter
Añadir/Editar	Cancelar	Alt+C	Escape
Búsqueda	Aceptar	Alt+A	Enter

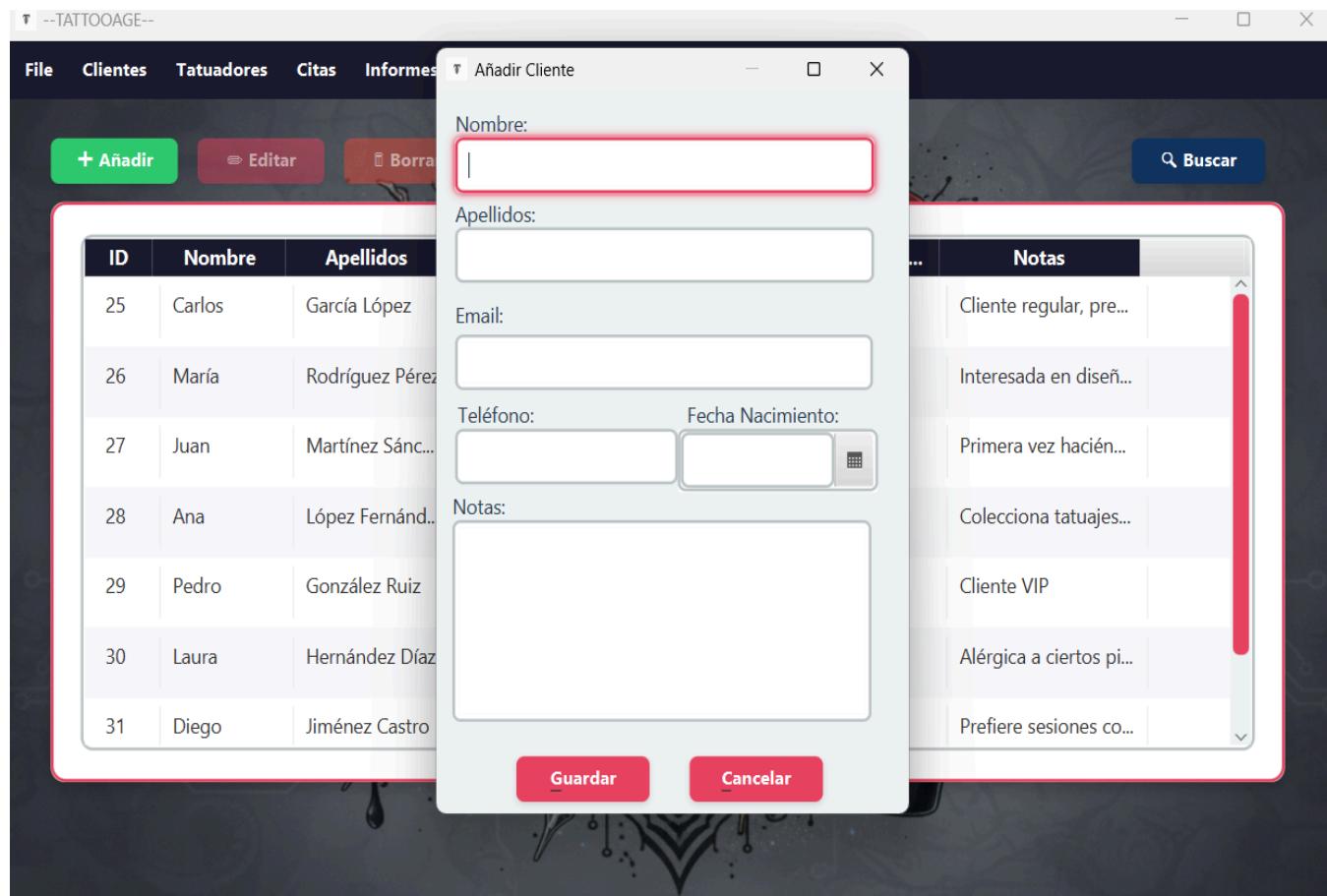
Ventana	Botón	Atajo Alt	Tecla especial
Búsqueda	Cancelar	Alt+C	Escape
Exit	Sí	Alt+S	Enter
Exit	No	Alt+N	Escape

## Capturas de Pantalla

- **Antes:** No funcionaban atajos de teclado



- **Después:** Funcionan Alt+letra, Enter y Escape



## FALLO 5: Botones sin Iconos

### Descripción del Problema

Los botones de las ventanas secundarias no tenían iconos, a diferencia de los botones de la ventana principal.

### Tipo de Prueba

#### PRUEBA DE DISEÑO/USABILIDAD

Las pruebas de diseño verifican la coherencia visual de la interfaz.

### Solución Implementada

Se añadieron emojis como iconos a todos los botones de las ventanas secundarias.

### Archivos Modificados

Archivo	Botones actualizados
ventanaExit.fxml	✓ Sí, ✗ No
ventanaAECiente.fxml	💾 Guardar, ✗ Cancelar
ventanaAETatuador.fxml	💾 Guardar, ✗ Cancelar
ventanaAECita.fxml	🔍 Buscar, 📁 Seleccionar, 🗑 Eliminar, 💾 Guardar, ✗ Cancelar
ventanaBCliente.fxml	✓ Aceptar, ✗ Cancelar
ventanaBTatuador.fxml	✓ Aceptar, ✗ Cancelar
ventanaBCita.fxml	✓ Aceptar, ✗ Cancelar

### Capturas de Pantalla

- **Antes:** Botones solo con texto



- **Después:** Botones con iconos y texto

The screenshot shows a modal window titled "Añadir Cliente" (Add Client). The form contains fields for Name, Surname, Email, Phone number, Date of Birth, and Notes. The "Guardar" (Save) button is highlighted with a red border.

Añadir Cliente

Nombre:

Apellidos:

Email:

Teléfono: Fecha Nacimiento:

Notas:

Guardar Cance...

FALLO 6: Falta de Tooltips

## Descripción del Problema

Los botones principales no mostraban información al pasar el ratón por encima (tooltips).

## Tipo de Prueba

### PRUEBA DE USABILIDAD

Los tooltips mejoran la experiencia del usuario proporcionando información contextual.

## Solución Implementada

Se añadieron tooltips a todos los botones de la ventana principal.

## Archivos Modificados

### ventanaPrincipal.fxml

```
<!-- Import añadido -->
<?import javafx.scene.control.Tooltip?>

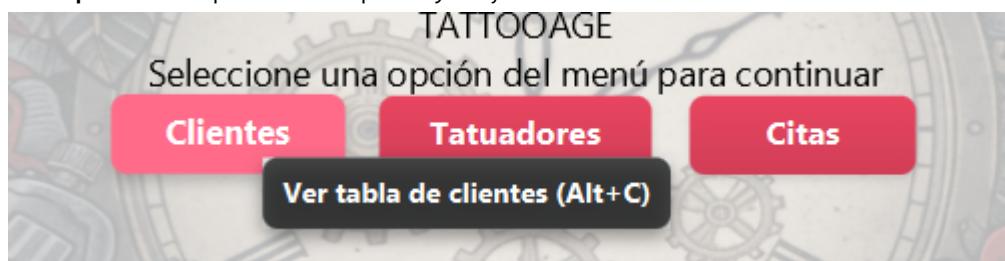
<!-- Ejemplo de botón con tooltip -->
<Button fx:id="btnAnadir" text="✚ _Añadir">
    <tooltip>
        <Tooltip text="Añadir nuevo registro (Alt+A)" />
    </tooltip>
</Button>
```

## Tabla de Tooltips Añadidos

Elemento	Tooltip
✚ Añadir	"Añadir nuevo registro (Alt+A)"
-pencil Editar	"Editar registro seleccionado (Alt+E)"
trash Borrar	"Borrar registro seleccionado (Alt+B)"
magnifying-glass Buscar	"Buscar registros (Alt+U)"
refresh Resetear filtro	"Quitar filtro de búsqueda (Alt+R)"
clients Clientes	"Ver tabla de clientes (Alt+C)"
tattoo-artists Tatuadores	"Ver tabla de tatuadores (Alt+T)"
citas Citas	"Ver tabla de citas (Alt+I)"
incrustado Incrustado	"Mostrar informe dentro de la aplicación (Alt+I)"
generar-informe Generar Informe	"Generar informe de citas por sala (Alt+G)"
informe-grafico Informe Gráfico	"Generar informe gráfico de citas (Alt+R)"

## Capturas de Pantalla

- **Antes:** Sin información al pasar el ratón
- **Después:** Tooltip con descripción y atajo de teclado



## FALLO 7: Botón VER Imagen Desplazado

### Descripción del Problema

El botón "Ver" en la tabla de citas para visualizar la imagen del diseño estaba desplazado y apenas visible.

### Tipo de Prueba

### PRUEBA DE DISEÑO/USABILIDAD

Las pruebas de diseño verifican que los elementos de la interfaz sean visibles y accesibles.

### Solución Implementada

Se ajustó el tamaño y alineación del botón y la columna.

### Archivos Modificados

#### miControlador.java

```
// ANTES
TableColumn<Cita, Void> colImagen = new TableColumn<>("Foto");
colImagen.setPrefWidth(80);
// ...
private final Button btn = new Button("Ver");

// DESPUÉS
TableColumn<Cita, Void> colImagen = new TableColumn<>("Foto");
colImagen.setPrefWidth(70);
colImagen.setMinWidth(70);
colImagen.setStyle("-fx-alignment: CENTER;");
// ...
private final Button btn = new Button("Ver");
{
    btn.setMinWidth(60);
    btn.setMaxWidth(60);
}

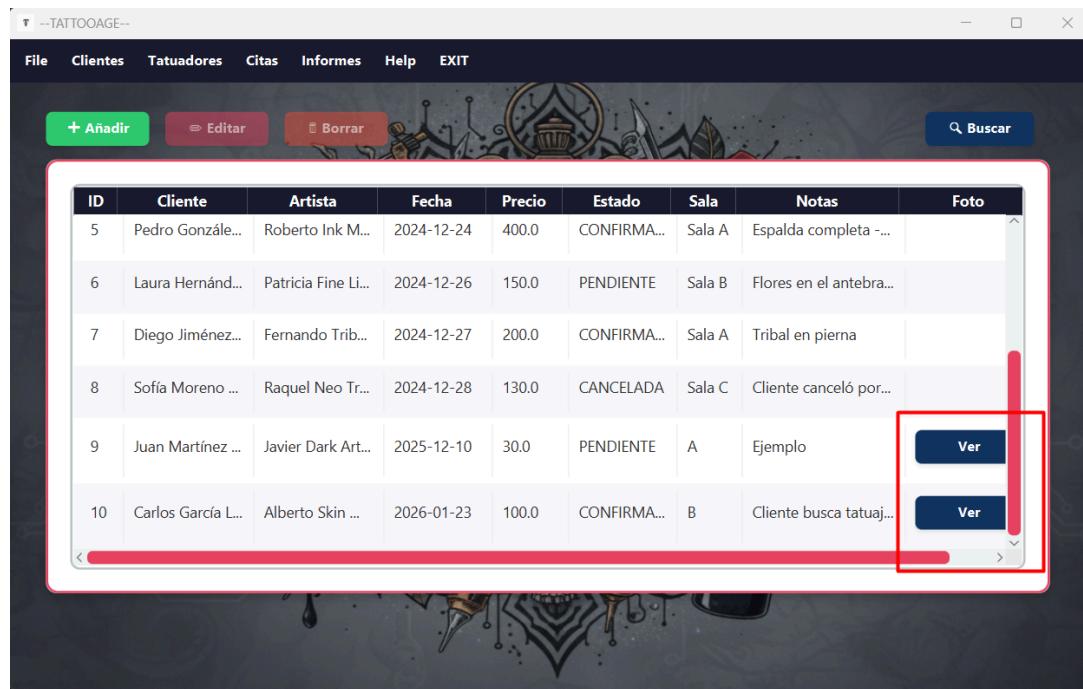
@Override
public void updateItem(Void item, boolean empty) {
    super.updateItem(item, empty);
    setAlignment(javafx.geometry.Pos.CENTER); // AÑADIDO
    // ... resto del código
}
```

## Cambios Realizados

Propiedad	Antes	Después
Texto botón	"Ver"	"  Ver"
Ancho columna	prefWidth(80)	prefWidth(70) + minWidth(70)
Ancho botón	Sin definir	minWidth(60) + maxWidth(60)
Alineación columna	Sin definir	CENTER
Alineación celda	Sin definir	Pos.CENTER

## Capturas de Pantalla

- Antes: Botón cortado o no visible



- Despues: Botón centrado y completamente visible



## Resumen de Archivos Modificados

### Archivos FXML

Archivo	Cambios
ventanaPrincipal.fxml	Labels error, mnemonics, tooltips
ventanaExit.fxml	Mnemonics, iconos, defaultButton/cancelButton
ventanaAECliente.fxml	Mnemonics, iconos, defaultButton/cancelButton
ventanaAETatuador.fxml	Mnemonics, iconos, defaultButton/cancelButton
ventanaAECita.fxml	Mnemonics, iconos, defaultButton/cancelButton
ventanaBCliente.fxml	Mnemonics, iconos, defaultButton/cancelButton
ventanaBTatuador.fxml	Mnemonics, iconos, defaultButton/cancelButton
ventanaBCita.fxml	Mnemonics, iconos, defaultButton/cancelButton

### Archivos Java

Archivo	Cambios
miControlador.java	Manejo errores BBDD, modality, iconos Alert, botón VER
ventanaAEClienteController.java	Iconos Alert
ventanaAETatuadorController.java	Iconos Alert
ventanaAECitaController.java	Iconos Alert
ventanaBClienteController.java	Iconos Alert
ventanaBTatuadorController.java	Iconos Alert

Archivo	Cambios
ventanaBCitaController.java	Iconos Alert
ErrorUtils.java	Iconos Alert

## Conclusión

Se han corregido un total de **7 fallos** identificados en las pruebas del primer trimestre:

- **2 fallos de SISTEMA** (robustez y funcionalidad)
- **5 fallos de DISEÑO/USABILIDAD**

Todas las correcciones mejoran la experiencia del usuario y la robustez de la aplicación, cumpliendo con los estándares de calidad esperados para una aplicación de gestión profesional.