

Tarea Módulo 2

**“Paneles y Layouts
como profesionales”**

JESÚS MACÍAS

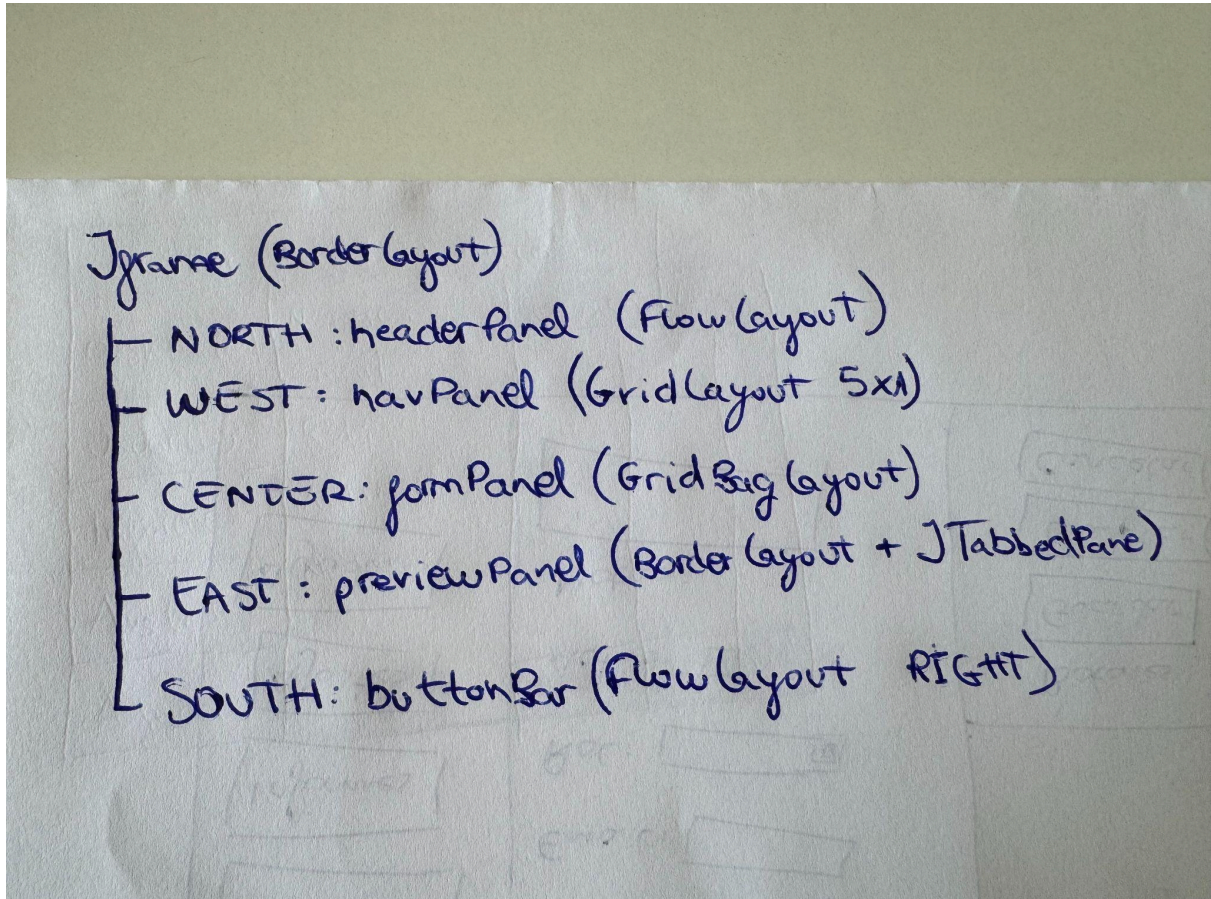


Planificación en papel

Gestor de usuarios			
Dashboard	Nombre:	Resumen	Logs
Usuarios	Email:		
Informes	Rol:		
Ajustes	Activo:		
Ayuda	Notas:		
		Cancelar	Limpiar guardar



Árbol de contenedores





Capturas de pantalla

Gestor de usuarios

Gestor de usuarios

Nombre:

Email:

Rol:

Activo: ☐

Notas:

Resumen Logs

Cancelar Limpiar Guardar

Gestor de usuarios

Gestor de usuarios

Nombre:

Email:

Rol:

Activo: ☐

Notas:

Resumen Logs

Cancelar Limpiar Guardar



Gestor de usuarios

Gestor de usuarios

Dashboard

Usuarios

Informes

Ajustes

Ayuda

Nombre:

Email:

Rol:

Activo: ☒

Notas:

Resumen

Logs

Confirmación

¿Guardar cambios?

Aceptar Cancelar

Cancelar

Limpiar

Guardar

Gestor de usuarios

Gestor de usuarios

Dashboard

Usuarios

Informes

Ajustes

Ayuda

Nombre:

Email:

Rol:

Activo: ☒

Notas:

Resumen

Logs

Message

Datos guardados (simulado).

OK

Cancelar

Limpiar

Guardar



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;

public class GestorUsuarios {
    private JFrame frame; 14 usages
    private JTextField tfNombre, tfEmail; 3 usages
    private JComboBox<String> cbRol; 3 usages
    private JCheckBox chActivo; 3 usages
    private JTextArea taNotas; 3 usages
    private JButton btnGuardar; 3 usages

    public GestorUsuarios() { 1 usage
        // JFrame principal
        frame = new JFrame( title: "Gestor de usuarios");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setSize( width: 1000, height: 600);
        frame.setLayout(new BorderLayout( hgap: 5, vgap: 5));

        // ----- Header -----
        JPanel headerPanel = new JPanel(new FlowLayout(FlowLayout.LEFT));
        JLabel lblHeader = new JLabel( text: "Gestor de usuarios");
        lblHeader.setFont(new Font( name: "Arial", Font.BOLD, size: 18));
        headerPanel.add(lblHeader);
        frame.add(headerPanel, BorderLayout.NORTH);
    }
}
```

```
// ----- Navegación -----
JPanel navPanel = new JPanel(new GridLayout( rows: 5, cols: 1, hgap: 0, vgap: 5));
String[] botones = {"Dashboard", "Usuarios", "Informes", "Ajustes", "Ayuda"};
for(String b : botones){
    JButton btn = new JButton(b);
    navPanel.add(btn);
}
frame.add(navPanel, BorderLayout.WEST);

// ----- Formulario -----
JPanel formPanel = new JPanel(new GridBagLayout());
GridBagConstraints gbc = new GridBagConstraints();
gbc.insets = new Insets( top: 5, left: 5, bottom: 5, right: 5);
gbc.anchor = GridBagConstraints.WEST;

// Nombre
gbc.gridx = 0; gbc.gridy = 0; gbc.weightx = 0; gbc.fill = GridBagConstraints.NONE;
formPanel.add(new JLabel( text: "Nombre:"), gbc);
tfNombre = new JTextField( columns: 20);
gbc.gridx = 1; gbc.weightx = 1; gbc.fill = GridBagConstraints.HORIZONTAL;
formPanel.add(tfNombre, gbc);
```



```
// Email
gbc.gridx = 0; gbc.gridy = 1; gbc.weightx = 0; gbc.fill = GridBagConstraints.NONE;
formPanel.add(new JLabel( text: "Email:"), gbc);
tfEmail = new JTextField( columns: 20);
gbc.gridx = 1; gbc.weightx = 1; gbc.fill = GridBagConstraints.HORIZONTAL;
formPanel.add(tfEmail, gbc);

// Rol
gbc.gridx = 0; gbc.gridy = 2; gbc.weightx = 0; gbc.fill = GridBagConstraints.NONE;
formPanel.add(new JLabel( text: "Rol:"), gbc);
cbRol = new JComboBox<>(new String[]{"Admin", "Editor", "Invitado"});
gbc.gridx = 1; gbc.weightx = 1; gbc.fill = GridBagConstraints.HORIZONTAL;
formPanel.add(cbRol, gbc);

// Activo
gbc.gridx = 0; gbc.gridy = 3; gbc.weightx = 0; gbc.fill = GridBagConstraints.NONE;
formPanel.add(new JLabel( text: "Activo:"), gbc);
chActivo = new JCheckBox();
gbc.gridx = 1; gbc.weightx = 1; gbc.fill = GridBagConstraints.HORIZONTAL;
formPanel.add(chActivo, gbc);
```

```
// Notas
gbc.gridx = 0; gbc.gridy = 4; gbc.weightx = 0; gbc.weighty = 0; gbc.fill = GridBagConstraints.NONE;
formPanel.add(new JLabel( text: "Notas:"), gbc);
taNotas = new JTextArea( rows: 5, columns: 20);
JScrollPane scrollNotas = new JScrollPane(taNotas);
gbc.gridx = 1; gbc.weightx = 1; gbc.weighty = 1; gbc.fill = GridBagConstraints.BOTH;
formPanel.add(scrollNotas, gbc);

frame.add(formPanel, BorderLayout.CENTER);

// ----- Previsualización -----
JPanel previewPanel = new JPanel(new BorderLayout());
previewPanel.setPreferredSize(new Dimension( width: 260, height: 0));
JTabbedPane tabbedPane = new JTabbedPane();

JTextArea taResumen = new JTextArea();
taResumen.setEditable(false);
tabbedPane.addTab( title: "Resumen", new JScrollPane(taResumen));

JTextArea taLogs = new JTextArea();
taLogs.setEditable(false);
tabbedPane.addTab( title: "Logs", new JScrollPane(taLogs));

previewPanel.add(tabbedPane, BorderLayout.CENTER);
frame.add(previewPanel, BorderLayout.EAST);
```



```
// ----- Botonera -----  
JPanel buttonBar = new JPanel(new FlowLayout(FlowLayout.RIGHT, 5, 5));  
JButton btnCancelar = new JButton( text: "Cancelar");  
JButton btnLimpiar = new JButton( text: "Limpiar");  
btnGuardar = new JButton( text: "Guardar");  
  
buttonBar.add(btnCancelar);  
buttonBar.add(btnLimpiar);  
buttonBar.add(btnGuardar);  
frame.add(buttonBar, BorderLayout.SOUTH);  
  
// ----- Eventos -----  
btnGuardar.addActionListener(this::abrirDialogConfirmacion);  
btnLimpiar.addActionListener( ActionEvent e -> limpiarFormulario());  
btnCancelar.addActionListener( ActionEvent e -> frame.dispose());  
  
// Mostrar ventana  
frame.setVisible(true);  
}
```

```
// Limpiar formulario  
private void limpiarFormulario() { 1 usage  
    tfNombre.setText("");  
    tfEmail.setText("");  
    cbRol.setSelectedIndex(0);  
    chActivo.setSelected(false);  
    taNotas.setText("");  
}
```




```
// Diálogo modal de confirmación
private void abrirDialogConfirmacion(ActionEvent e) { 1usage
    JDialog dialog = new JDialog(frame, title: "Confirmación", modal: true);
    dialog.setLayout(new BorderLayout( hgap: 10, vgap: 10));
    JLabel lbl = new JLabel( text: "¿Guardar cambios?");
    lbl.setBorder(BorderFactory.createEmptyBorder( top: 10, left: 10, bottom: 0, right: 10));
    dialog.add(lbl, BorderLayout.CENTER);

    JPanel buttons = new JPanel(new FlowLayout(FlowLayout.RIGHT));
    JButton aceptar = new JButton( text: "Aceptar");
    JButton cancelar = new JButton( text: "Cancelar");
    buttons.add(aceptar);
    buttons.add(cancelar);
    dialog.add(buttons, BorderLayout.SOUTH);

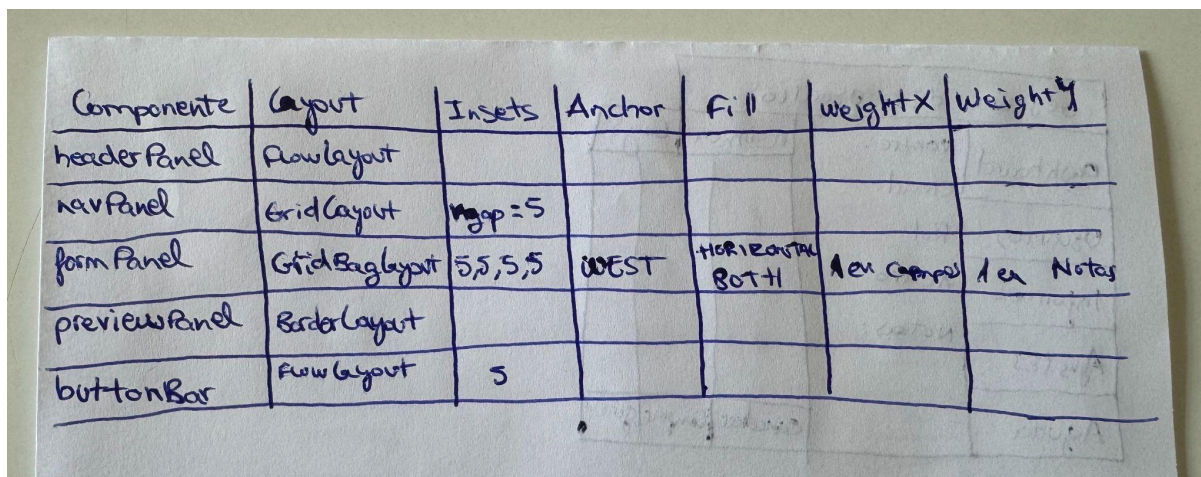
    aceptar.addActionListener( ActionEvent ev -> {
        JOptionPane.showMessageDialog(frame, message: "Datos guardados (simulado).");
        dialog.dispose();
    });

    cancelar.addActionListener( ActionEvent ev -> dialog.dispose());

    dialog.pack();
    dialog.setLocationRelativeTo(frame);
    dialog.setVisible(true);
}

public static void main(String[] args) {
    SwingUtilities.invokeLater(GestorUsuarios::new);
}
```

Mini tabla con las propiedades clave usadas



Componente	Layout	Insets	Anchor	Fill	weightX	WeightY
headerPanel	FlowLayout					
navPanel	GridLayout	gap=5				
formPanel	GridBagLayout	5,5,5,5	WEST	HORIZONTAL BOTH	1 en Campos	1 en Notas
previewPanel	BorderLayout					
buttonBar	FlowLayout	5				

CONCLUSIÓN

Esta práctica me ha servido para entender mejor cómo se usan los layouts en Java, al principio cuesta un poco porque cada uno funciona distinto, pero viendo cómo se combinan en la ventana se entiende mucho mejor. El que más me ha costado ha sido el **GridBagLayout**, pero también es el más útil porque deja que el formulario se adapte cuando cambias el tamaño de la ventana. Con el **BorderLayout**, **FlowLayout** y **GridLayout** fue más sencillo, y también he visto que es importante planificar primero (wireframe y árbol de paneles), porque así después en el código es más fácil. En general creo que la actividad me ayudó a organizar mejor las interfaces y a perderle miedo a los layouts de Java.