

✓ Resumo do conteúdo explicado

♦ Eventos como gatilhos de ações

- Quando o usuário interage com a interface (clique, digitação, seleção), um **evento é disparado**.
- Esse evento pode:
 - **coletar dados de campos** (`getText()` em `JTextField`)
 - **preencher ou limpar campos**
 - **acessar banco de dados** (ex: buscar por CPF)
 - **validar credenciais** de login

♦ Componentes gráficos interagindo entre si

Exemplo prático:

- Usuário digita CPF em um `JTextField`
- Clica em um botão `JButton`
- Um `ActionListener` executa a busca no banco de dados
- Os dados retornados preenchem outros `JTextField` (nome, telefone, etc.)

🔧 Exemplo adaptado com `JComboBox` preenchido no construtor (Código 57):

```
cmbSexo = new JComboBox<>();
cmbSexo.addItem("Masculino");
cmbSexo.addItem("Feminino");
cmbSexo.addItem("Outro");
painel.add(cmbSexo);
```

Ou, de forma mais enxuta:

```
String[] sexos = {"Masculino", "Feminino", "Outro"};
```

```
cmbSexo = new JComboBox<>(sexos);
painel.add(cmbSexo);
```

Quadro 10 – Métodos úteis em componentes GUI

Esses métodos te ajudam a controlar o comportamento dos componentes:

Método	Utilidade
<code>getName()</code> / <code>setName(String)</code>	Útil para identificar componentes
<code>getParent()</code>	Descobre em que <code>JPanel</code> ou <code>JFrame</code> o componente está
<code>setVisible(boolean)</code>	Mostra ou esconde componentes dinamicamente
<code>isShowing()</code>	Verifica se o componente está visível
<code>isEnabled()</code> / <code>setEnabled(boolean)</code>	Ativa ou desativa campos ou botões
<code>setToolTipText(String)</code>	Define a mensagem que aparece ao passar o mouse
<code>setPreferredSize(Dimension)</code>	Define o tamanho preferido de exibição

Exemplo completo prático: Tela de login

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

public class TelaLogin extends JFrame {
    private JTextField txtUsuario;
    private JPasswordField txtSenha;
    private JButton btnLogin;

    public TelaLogin() {
        setTitle("Login");
        setSize(300, 180);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setLayout(new GridLayout(3, 2, 10, 10));

        add(new JLabel("Usuário:"));
```

```

txtUsuario = new JTextField();
add(txtUsuario);

add(new JLabel("Senha:"));
txtSenha = new JPasswordField();
add(txtSenha);

btnLogin = new JButton("Entrar");
add(btnLogin);

// Espaço em branco para alinhamento
add(new JLabel(""));

// Evento de ação no botão
btnLogin.addActionListener(e -> {
    String usuario = txtUsuario.getText();
    String senha = new String(txtSenha.getPassword());

    if (usuario.equals("admin") && senha.equals("123")) {
        JOptionPane.showMessageDialog(null, "Login bem-sucedido!");
    } else {
        JOptionPane.showMessageDialog(null, "Usuário ou senha incorretos.");
    }
});
}

public static void main(String[] args) {
    new TelaLogin().setVisible(true);
}
}

```

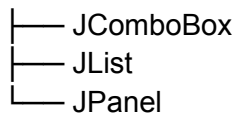
Hierarquia dos Componentes Swing (Figura 16)

Embora você tenha mencionado uma figura (provavelmente um diagrama visual), aqui está a estrutura **conceitual da hierarquia dos componentes Swing**, simplificada:

```

java.lang.Object
├── java.awt.Component
│   ├── java.awt.Container
│   │   ├── javax.swing.JComponent
│   │   │   ├── JLabel
│   │   │   ├── JTextField
│   │   │   ├── JButton
│   │   │   └── JCheckBox

```



💡 **Todos os componentes Swing** herdam de `JComponent`, o que significa que todos eles têm métodos como `setVisible()`, `setEnabled()`, `setToolTipText()`, etc.

🧱 Quadro 11 – Componentes de Interface Gráfica Swing

Aqui está uma versão revisada com exemplos de uso e complementos:

Componente	Descrição	Exemplo de uso
<code>JLabel</code>	Exibe um texto fixo ou um ícone. Não editável.	<code>new JLabel("Nome:")</code>
<code>JTextField</code>	Campo para entrada de texto em uma linha.	<code>new JTextField(20)</code>
<code>JButton</code>	Botão que dispara ações via <code>ActionListener</code> .	<code>new JButton("Enviar")</code>
<code>JCheckBox</code>	Caixa de seleção (ligado/desligado).	<code>new JCheckBox("Aceito os termos")</code>
<code>JComboBox</code>	Lista suspensa para escolher um item.	<code>new JComboBox<>(new String[]{"Masculino", "Feminino"})</code>
<code>JList</code>	Lista de múltiplas seleções ou única.	<code>new JList<>(new String[]{"Item 1", "Item 2"})</code>
<code>JPanel</code>	Painel que agrupa e organiza componentes (usado para layout).	<code>JPanel panel = new JPanel(); panel.add(label);</code>

📌 Exemplo de layout com esses componentes:

```
JPanel panel = new JPanel();  
panel.setLayout(new GridLayout(4, 2)); // 4 linhas, 2 colunas
```

```
panel.add(new JLabel("Nome:"));
panel.add(new JTextField(15));
```

```
panel.add(new JLabel("Sexo:"));
panel.add(new JComboBox<>(new String[]{"Masculino", "Feminino"}));
```

```
panel.add(new JCheckBox("Aceito os termos"));
```

```
panel.add(new JButton("Enviar"));
```