

4. CAPTURAS

PantallaPrincipal Source

```
private boolean validarValores(String name, String password) {
    if (name.isBlank() || password.isBlank()) {
        JOptionPane.showMessageDialog(parentComponent.this, message:"El nombre o la contraseña no pueden estar vacios", title:"Error", messageType: JOptionPane.ERROR_MESSAGE);
        return false;
    }
    return true;
}
```

```
private boolean nameExist(String name) {
    if (logs.containsKey(key: name)) {
        return true;
    }
    return false;
}
```

```
public boolean anadirUsuario(String name, String pass) {
    if (validarValores(name, password: pass)) {
        if (!nameExist(name)) {
            logs.put(key: name, value: pass);
            return true;
        } else {
            JOptionPane.showMessageDialog(parentComponent.this, message:"El nombre ya existe.", title:"Error", messageType: JOptionPane.ERROR_MESSAGE);
        }
    }
    return false;
}
```

```
private boolean validarUsuario() {
    String name = jTextFieldUser.getText();
    char pass[] = jPasswordFieldPass.getPassword();
    String password = new String(value: pass);
    if (validarValores(name, password)) {
        String passw = logs.get(key: name);
        if (nameExist(name) && passw.equals(anObject: password)) {
            return true;
        } else {
            JOptionPane.showMessageDialog(parentComponent.this, message:"El nombre o la contraseña no existen", title:"Error", messageType: JOptionPane.ERROR_MESSAGE);
        }
    }
    return false;
}
```

```
private void jButtonLogActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if (validarUsuario()) {
        JOptionPane.showMessageDialog(parentComponent.this, message:"Has entrado correctamente", title:"Login correcto", messageType: JOptionPane.INFORMATION_MESSAGE);
    }
}

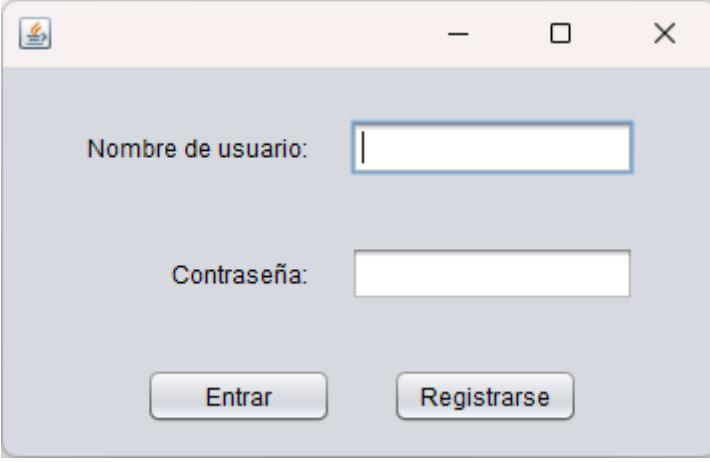
private void jButtonRegActionPerformed(java.awt.event.ActionEvent evt) {
    new PantallaRegistro(parent: this, modal: true).setVisible(true);
}
}
```

PantallaRegistro Source

```
PantallaPrincipal firsScreen;
public PantallaRegistro(java.awt.Frame parent, boolean modal) {
    super(owner: parent, modal);
    firsScreen = (PantallaPrincipal) parent;
    initComponents();
}
```

```
private void jButtonRegActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String name = jTextFieldUser.getText();
    String password = new String(value: jPasswordFieldPass.getPassword());
    if(firsScreen.anadirUsuario(name, pass: password)){
        this.dispose();
    }
}
```

PantallaPrincipal Design

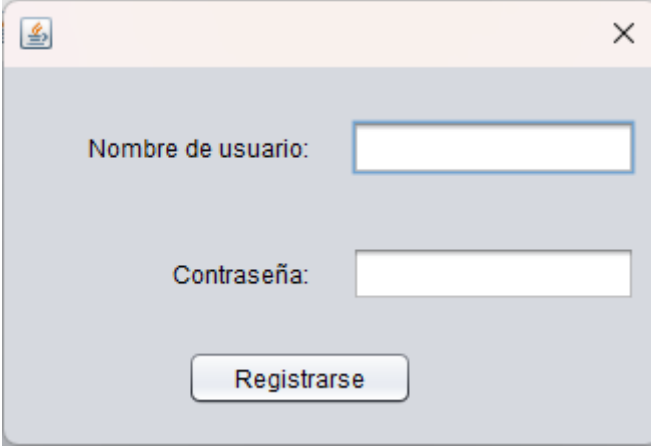


A screenshot of a software window titled "PantallaPrincipal Design". The window has a light gray background and a title bar with a small icon on the left and standard minimize, maximize, and close buttons on the right. The main content area contains two text input fields. The first field is labeled "Nombre de usuario:" and has a blue border. The second field is labeled "Contraseña:" and is empty. Below the fields are two buttons: "Entrar" and "Registrarse".

Nombre de usuario:

Contraseña:

PantallaRegistro Design



A screenshot of a software window titled "PantallaRegistro Design". The window has a light gray background and a title bar with a small icon on the left and a close button on the right. The main content area contains two text input fields. The first field is labeled "Nombre de usuario:" and has a blue border. The second field is labeled "Contraseña:" and is empty. Below the fields is a single button: "Registrarse".

Nombre de usuario:

Contraseña: