

| | | | | | |
|------------------------------|---|-----------------|---|--------------------------------------|----------------|
| Nombre de la práctica | UNIDAD 3. CLASES Y OBJETOS | | | No. | 1 |
| Asignatura: | Programación Orientada a Objetos | Carrera: | Ingeniería en Sistemas Computacionales | Duración de la práctica (Hrs) | 8 horas |

NOMBRE DEL ALUMNO: Jesus Navarrete Martínez

GRUPO: 3203

I. Competencia(s) específica(s):

Hacer uso de métodos y atributos a través de la herencia simple aplicada en la interfaz grafica.

Encuadre con CACEI: Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en esta práctica.

| No. atributo | Atributos de egreso del PE que impactan en la asignatura | Criterios de desempeño | |
|--------------|--|------------------------|--|
| 2 | El estudiante diseñará esquemas de trabajo y procesos, usando metodologías congruentes en la resolución de problemas de ingeniería en sistemas computacionales | 1 | Identifica metodologías y procesos empleados en la resolución de problemas |
| | | 2 | Diseña soluciones a problemas, empleando metodologías apropiadas al área |
| 3 | El estudiante plantea soluciones basadas en tecnologías empleando su juicio ingenieril para valorar necesidades, recursos y resultados esperados. | 1 | Emplea los conocimientos adquiridos para el desarrollar soluciones |
| | | 2 | Analiza y comprueba resultados |

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

Actividades en aula de clases y en equipo personal

III. Material empleado:

- Equipo de cómputo
- Neat beans



IV. Desarrollo de la práctica:

Unidad 3

Primer practica:

Código de la practica: Encriptador

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Encriptadorr.java x
Source History
1 package encriptadorr;
2 //paquete
3 import java.awt.Container;
4 import java.awt.BorderLayout;
5 import java.awt.Color;
6 import java.awt.Font;
7 import java.awt.GridLayout;
8 import java.awt.event.ActionEvent;
9 import java.awt.event.ActionListener;
10 import javax.swing.JFrame;
11 import javax.swing.JPanel;
12 import javax.swing.JButton;
13 import javax.swing.JLabel;
14 import javax.swing.JTextField;
15 import javax.swing.SwingConstants;
16 //clase
17 public class Encriptadorr extends JFrame{
18 //Atributos
19 private JTextField entrada;
20 private JLabel titulo, entradaLbl, salida, salidaCadena, limp;
21 private JButton encriptar, desencriptar, limpiar, salir;
22 private JPanel central, norte, sur;
23 int l,i;
24 String encriptado,desencriptado;
25
26 //Constructor: es un constructor publico sin argumentos
27 public Encriptadorr(){
28     Eventos manejador=new Eventos();
29     Container contenedor = getContentPane();
30     contenedor.setLayout(new BorderLayout());
31     central = new JPanel(new GridLayout(3,2));
32     entrada = new JTextField(40);
33     norte = new JPanel (new GridLayout(1,1));
34     titulo = new JLabel ("EJERCICIO PARA ENCRIPITAR", SwingConstants.CENTER);
35     entradaLbl = new JLabel("Texto a Encriptar", SwingConstants.CENTER);
36     salida = new JLabel("ENCRIPTAR", SwingConstants.CENTER);
37     salidaCadena = new JLabel ("DESENCRIPTAR", SwingConstants.CENTER);
38     encriptar = new JButton("ENCRIPTAR");
39     encriptar.addActionListener(manejador);
40     desencriptar = new JButton("DESENCRIPTAR");
41     desencriptar.addActionListener(manejador);
42     sur= new JPanel(new GridLayout(1,2));
43     limpiar= new JButton("Limpiar");
44     limpiar.addActionListener(manejador);
45     salir=new JButton("Salir");
46     salir.addActionListener(manejador);
47 }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Encriptadorr.java x
Source History
51 norte.add(titulo);
52 central.add(entradaLbl);
53 central.add(entrada);
54 sur.add(limpiar);
55 sur.add(salir);
56 entradaLbl.setFont(new Font("Cooper Black",Font.PLAIN, 16));
57 entradaLbl.setForeground(Color.orange);
58 central.add(encriptar);
59 central.add(salida);
60 salida.setFont(new Font("Cooper Black",Font.PLAIN, 16));
61 salida.setForeground(Color.green);
62 central.add(desencriptar);
63 central.add(salidaCadena);
64 salidaCadena.setFont(new Font("Cooper Black",Font.PLAIN, 16));
65 salidaCadena.setForeground(Color.blue);
66 sur.add(limpiar);
67 sur.add(salir);
68
69 contenedor.add(norte, BorderLayout.NORTH);
70
71 titulo.setFont(new Font("Cooper Black",Font.PLAIN, 18));
72 titulo.setForeground(Color.black);
73 norte.setBackground(Color.green);
74
75
76 contenedor.add(central, BorderLayout.CENTER);
77 contenedor.add(sur, BorderLayout.SOUTH);
78
79 setSize(400,300); // pixeles, está es la mamá.
80 setVisible(true); // se define com un valor booleano.
81 } // termina el constructor.
82
83 public static void main(String args[]){
84     Encriptadorr ventana = new Encriptadorr();
85     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
86 }
87 public void activar(){
88     char array[]=entrada.getText().toCharArray();
89     //arreglo que permite caracteres
90     l=array.length;
91     for(i=0;i<array.length;i++){
92         array[i]=(char) (array[i]+(char)l);
93         //que la posicion del arreglo se igual al mismo valor para no perder el valor original
94     }
95     encriptado=String.valueOf(array);
96     salida.setText(encriptado);
97 }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Encriptadorr.java x
Source History
98 } //cierra metodo
99 public void desactivar() {
100     char array[]=salida.getText().toCharArray();
101     l=array.length;
102     for(i=0;i<array.length;i++){
103         array[i]=(char) (array[i]-(char)l);
104     }
105     desencriptado=String.valueOf(array);
106     salidaCadena.setText(desencriptado);
107 } //cierra metodo
108
109 public void borrar() {
110     entrada.setText("");
111     salida.setText("");
112     salidaCadena.setText("");
113 }
114
115
116 class Eventos implements ActionListener{
117     @Override
118     public void actionPerformed(ActionEvent ev){
119         if(ev.getSource()==encriptar){
120             activar();
121         }
122
123         if(ev.getSource()==desencriptar){
124             desactivar();
125         }
126
127         if(ev.getSource()==limpiar){
128             borrar();
129         }
130
131         if(ev.getSource()==salir){
132             System.exit(0);
133         }
134     }
135 }
136
137
138
139
140
```

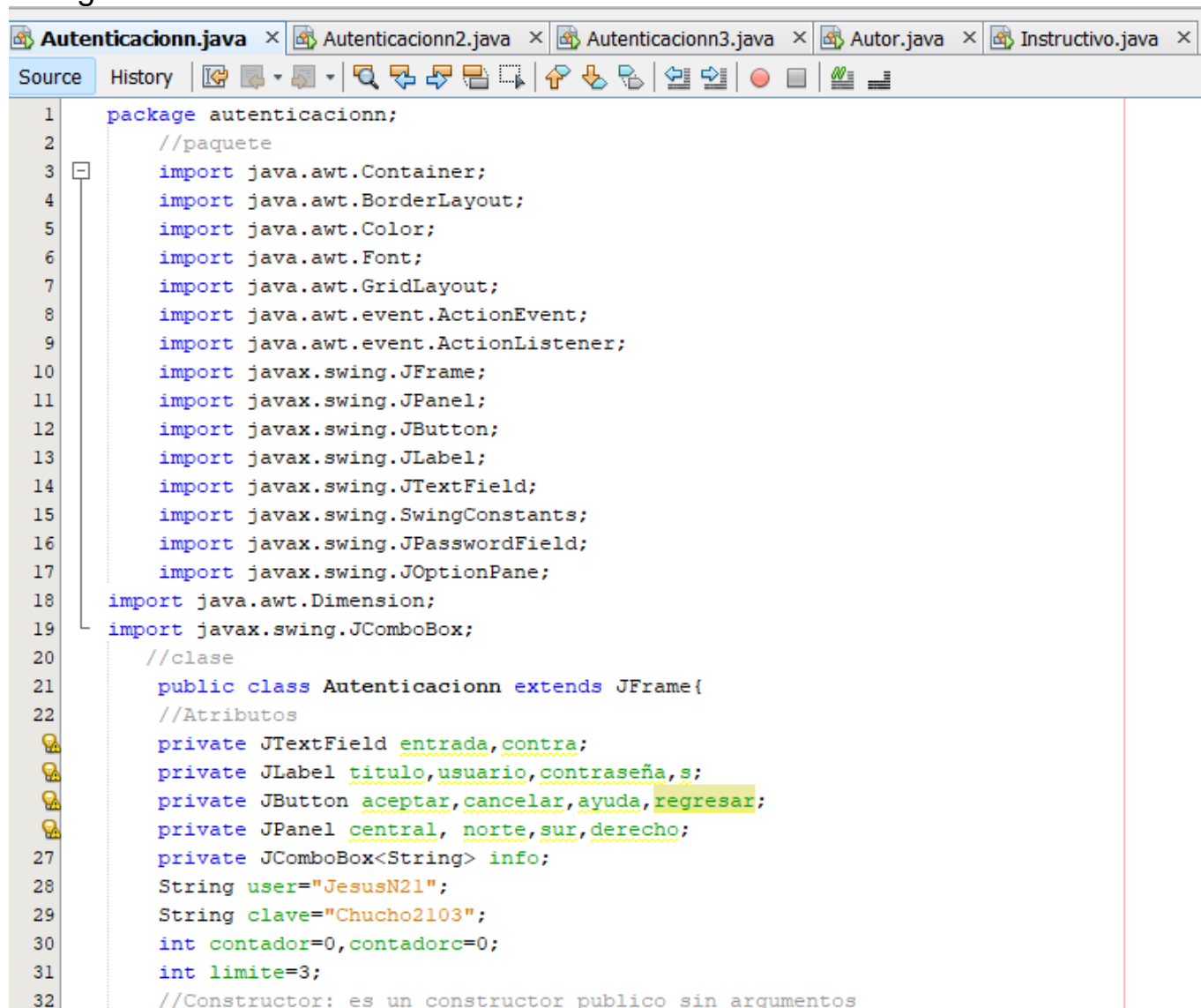


Resultado de correr el código Encriptador :

| EJERCICIO PARA ENCRIPtar | |
|---------------------------------|---------------------|
| Texto a Encriptar | |
| ENCRIPtar | ENCRIPtar |
| DESENCRIPTAR | DESENCRIPTAR |
| Limpiar | Salir |

código del proyecto Interfaz Gráfica:

Código de la ventana 1:



```
1 package autenticacionn;
2 //paquete
3 import java.awt.Container;
4 import java.awt.BorderLayout;
5 import java.awt.Color;
6 import java.awt.Font;
7 import java.awt.GridLayout;
8 import java.awt.event.ActionEvent;
9 import java.awt.event.ActionListener;
10 import javax.swing.JFrame;
11 import javax.swing.JPanel;
12 import javax.swing.JButton;
13 import javax.swing.JLabel;
14 import javax.swing.JTextField;
15 import javax.swing.SwingConstants;
16 import javax.swing.JPasswordField;
17 import javax.swing.JOptionPane;
18 import java.awt.Dimension;
19 import javax.swing.JComboBox;
20 //clase
21 public class Autenticacionn extends JFrame{
22 //Atributos
23 private JTextField entrada,contra;
24 private JLabel titulo,usuario,contraseña,s;
25 private JButton aceptar,cancelar,ayuda,regresar;
26 private JPanel central,norte,sur,derecho;
27 private JComboBox<String> info;
28 String user="JesusN21";
29 String clave="Chucho2103";
30 int contador=0,contadorc=0;
31 int limite=3;
32 //Constructor: es un constructor publico sin argumentos
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
33 public Autenticacionn(){
34     Eventos manejador=new Eventos();
35     Container contenedor = getContentPane();
36     contenedor.setLayout(new BorderLayout());
37     central = new JPanel(new GridLayout(2,2));
38     norte = new JPanel (new GridLayout(1,2));
39     derecho=new JPanel();
40     titulo = new JLabel ("Verificador de identidad", SwingConstants.CENTER);
41     s=new JLabel("<html>TIENES DUDAS?<br/>Por favor selececciona <br/>"
42         + "la informacion <br/>que deseas conocer: <html>",SwingConstants.NORTH_EAST);
43     s.setVisible(false);
44     String[] elementos1 = {"Informacion del autor","Instructivo"};
45     info= new JComboBox<>(elementos1);
46     info.setEditable(true);
47     ayuda= new JButton("?");
48     ayuda.setPreferredSize(new Dimension(45, 45));
49     ayuda.addActionListener(manejador);
50     regresar=new JButton("Volver al inicio");
51     regresar.setVisible(false);
52     regresar.addActionListener(manejador);
53     usuario= new JLabel ("Ingrese su usuario",SwingConstants.CENTER);
54     contraseña= new JLabel ("Ingrese su contraseña",SwingConstants.CENTER);
55     entrada=new JTextField(40);
56     contra= new JPasswordField(40);
57     sur= new JPanel(new GridLayout(1,1));
58     aceptar= new JButton("Aceptar");
59     aceptar.addActionListener(manejador);
60     cancelar= new JButton("Cancelar");
61     cancelar.addActionListener(manejador);
62     norte.add(titulo);
63     derecho.add(ayuda);
64     derecho.add(s);
65     derecho.add(info);
66     info.setVisible(false);
```



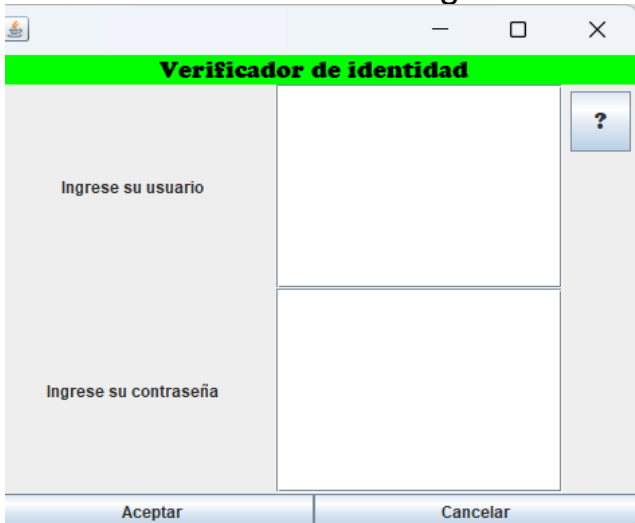
```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History

67     central.add(usuario);
68     central.add(entrada);
69     central.add(contraseña);
70     central.add(contra);
71     sur.add(aceptar);
72     sur.add(cancelar);
73     contenedor.add(norte, BorderLayout.NORTH);
74     contenedor.add(derecho, BorderLayout.EAST);
75     contenedor.add(central, BorderLayout.CENTER);
76     contenedor.add(sur, BorderLayout.SOUTH);
77     titulo.setFont(new Font("Cooper Black", Font.PLAIN, 18));
78     ayuda.setFont(new Font("Cooper Black", Font.PLAIN, 18));
79     titulo.setForeground(Color.black);
80     norte.setBackground(Color.green);
81     setSize(500,400); // pixeles, está es la mamá.
82     setVisible(true); // se define com un valor booleano.
83     info.addActionListener(e -> {
84         String opcionSeleccionada = (String) info.getSelectedItem();
85         if (opcionSeleccionada.equals("Informacion del autor")) {
86             Autor cuartaVentana= new Autor();
87             cuartaVentana.setVisible(true);
88             this.setVisible(false);
89         } else if(opcionSeleccionada.equals("Instructivo")) {
90             Instructivo quintaVentana= new Instructivo();
91             quintaVentana.setVisible(true);
92             this.setVisible(false);
93         }
94     });
95 } // termina el constructor.
96 public static void main(String args[]){
97     Autenticacionn ventana = new Autenticacionn();
98     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
99 }
100 public void borrar(){
101     entrada.setText("");
102     contra.setText("");
103 }
```



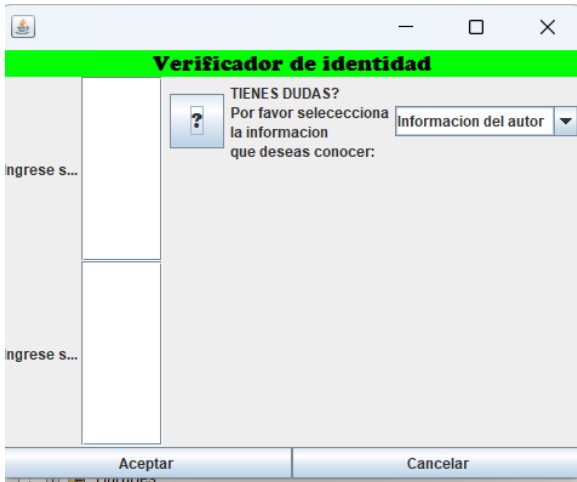

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History [Icons]
103
104     public void traer(){
105         Autenticacionn ven= new Autenticacionn();
106         ven.setVisible(true);
107         this.setVisible(false);
108     }
109     public void verificarUsuario(){
110         if(user.equals(entrada.getText()) && clave.equals(contra.getText())){
111             Autenticacionn2 segundaVentana= new Autenticacionn2();
112             segundaVentana.setVisible(true);
113             this.setVisible(false);
114         }else{
115             JOptionPane.showMessageDialog(null, "Contraseña o usuario incorrecto");
116             contador++;
117             entrada.setText("");
118             contra.setText("");
119         }
120         if(contador==limite){
121             System.exit(0);
122         }
123     }
124     class Eventos implements ActionListener{
125         @Override
126         public void actionPerformed(ActionEvent ev) {
127             if (ev.getSource() == cancelar) {
128                 borrar();
129             }
130             if (ev.getSource() == aceptar) {
131                 verificarUsuario();
132             }
133             if (ev.getSource() == ayuda) {
134                 s.setVisible(true);
135                 regresar.setVisible(true);
136                 info.setVisible(true);
137                 info.setPopupVisible(true);
138                 contadorc++;
139                 if (contadorc==2) {
140                     traer();
141                 }
142             }
143         }
144     }
145 }
146
147 }
148
```

Resultado al correr el código 1:



The screenshot shows a window titled "Verificador de identidad" with a green header bar. The main area is divided into two sections: "Ingrese su usuario" and "Ingrese su contraseña". There is a question mark button in the top right corner. At the bottom, there are "Aceptar" and "Cancelar" buttons.

Resultado al accionar el botón “?”:



The screenshot shows the same window after clicking the question mark button. The text "TIENES DUDAS?" is displayed, followed by "Por favor seleccione la información que desea conocer:". A dropdown menu is open, showing "Información del autor". The "Aceptar" and "Cancelar" buttons are still at the bottom.



Código de la ventana 2:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
1 package autenticacionn;
2 //Paquetes
3 import java.awt.Container;
4 import java.awt.BorderLayout;
5 import java.awt.Color;
6 import java.awt.Dimension;
7 import java.awt.Font;
8 import java.awt.GridLayout;
9 import java.awt.event.ActionEvent; //Acciones
10 import java.awt.event.ActionListener; //Detecta
11 import java.util.HashSet;
12 import javax.swing.JFrame;
13 import javax.swing.JPanel; //Manteles
14 import javax.swing.JButton;
15 import javax.swing.JLabel;
16 import javax.swing.JTextField;
17 import javax.swing.JComboBox;
18 import javax.swing.SwingConstants;
19 import javax.swing.JOptionPane;
20 //Clase
21 public class Autenticacionn2 extends JFrame implements Validacion {
22     //Atributos
23     private JTextField periodo;
24     private JLabel texto, ciclo, semestre, asignatura;
25     private JButton aceptar, cancelar, confirmar, confirmar2;
26     private JPanel central, run;
27     private JPanel inicio;
28     private JPanel ultimo;
29     private JComboBox<String> materia, sem;
30
31     //Constructor
32
33     public Autenticacionn2 () {
34         Eventos manejador=new Eventos();
35         Container contenedor=getContentPane();
36         contenedor.setLayout(new BorderLayout());
37         central=new JPanel(new GridLayout(3,2));
38         inicio=new JPanel(new GridLayout(1,1));
39         ultimo=new JPanel(new GridLayout(1,1));
40         run=new JPanel();
41         inicio.setBackground(Color.green);
42         periodo =new JTextField(10);
43         ciclo=new JLabel("PERIODO");
44         semestre=new JLabel("SEMESTRE");
45         semestre.setVisible(false);
46         String[] elementos1 = {"Semestre 1", "Semestre 2", "Semestre 3", "Semestre 4",
```



```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instruc
Source History
48         "Semestre 5", "Semestre 6", "Semestre 7", "Semestre 8");
49     sem= new JComboBox<>(elementos1);
50     sem.setVisible(false);
51
52     asignatura = new JLabel("MATERIA");
53     asignatura.setVisible(false);
54     String[] elementos2 = {"Fundamentos de programacion"};
55     materia= new JComboBox<>(elementos2);
56     materia.setVisible(false);
57     texto = new JLabel ("BIENVENID@", SwingConstants.CENTER);
58     aceptar=new JButton ("Aceptar");
59     aceptar.addActionListener (manejador);
60     cancelar=new JButton ("Cancelar");
61     cancelar.addActionListener (manejador);
62     confirmar= new JButton("confirmar");
63     confirmar.addActionListener (manejador);
64     confirmar.setPreferredSize(new Dimension(45, 45));
65     confirmar2= new JButton("confirmar");
66     confirmar2.addActionListener (manejador);
67     confirmar2.setPreferredSize(new Dimension(45, 45));
68     confirmar2.setVisible(false);
69     inicio.add(texto);
70     run.add(confirmar);
71     central.add(ciclo);
72     central.add(periodo);
73     central.add(confirmar);
74     central.add(semestre);
75     central.add(sem);
76     central.add(confirmar2);
77     central.add(asignatura);
78     central.add(materia);
79     ultimo.add(aceptar);
80     ultimo.add(cancelar);
81     contenedor.add(run, BorderLayout.EAST);
82     contenedor.add(central, BorderLayout.CENTER);
83     contenedor.add(inicio, BorderLayout.NORTH);
84     contenedor.add(ultimo, BorderLayout.SOUTH);
85     setSize(500,400);
86     setVisible(true);
87     texto.setForeground(Color.BLACK);
88     texto.setFont(new Font ("Cooper Black", Font.PLAIN,24));
89     sem.addActionListener(e -> {
90         String opcionSeleccionada = (String) sem.getSelectedItem();
91         if (opcionSeleccionada.equals("Semestre 1")) {
92
93             String[] subopciones1 = {"Fundamentos de programacion"};
94             materia.removeAllItems();

```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instru
Source History
95     for (String subopcion : subopciones1) {
96         materia.addItem(subopcion);
97     }
98 } else if (opcionSeleccionada.equals("Semestre 2")) {
99     String[] subopciones2 = {"Programacion orientada a"};
100    materia.removeAllItems();
101    for (String subopcion : subopciones2) {
102        materia.addItem(subopcion);
103    }
104 } else if (opcionSeleccionada.equals("Semestre 3")) {
105     String[] subopciones2 = {"Sistemas Operativos"};
106     materia.removeAllItems();
107     for (String subopcion : subopciones2) {
108         materia.addItem(subopcion);
109     }
110 } else if (opcionSeleccionada.equals("Semestre 4")) {
111     String[] subopciones2 = {"Arquitectura de computadoras"};
112     materia.removeAllItems();
113     for (String subopcion : subopciones2) {
114         materia.addItem(subopcion);
115     }
116 } else if (opcionSeleccionada.equals("Semestre 5")) {
117     String[] subopciones2 = {"Redes de computadoras"};
118     materia.removeAllItems();
119     for (String subopcion : subopciones2) {
120         materia.addItem(subopcion);
121     }
122 } else if (opcionSeleccionada.equals("Semestre 6")) {
123     String[] subopciones2 = {"Programacion web"};
124     materia.removeAllItems();
125     for (String subopcion : subopciones2) {
126         materia.addItem(subopcion);
127     }
128 } else if (opcionSeleccionada.equals("Semestre 7")) {
129     String[] subopciones2 = {"Internet de las cosas"};
130     materia.removeAllItems();
131     for (String subopcion : subopciones2) {
132         materia.addItem(subopcion);
133     }
134 }
135 else {
136     String[] subopciones3 = {"Inteligencia artificial"};
137     materia.removeAllItems();
138     for (String subopcion : subopciones3) {
139         materia.addItem(subopcion);
140     }
141 }
```

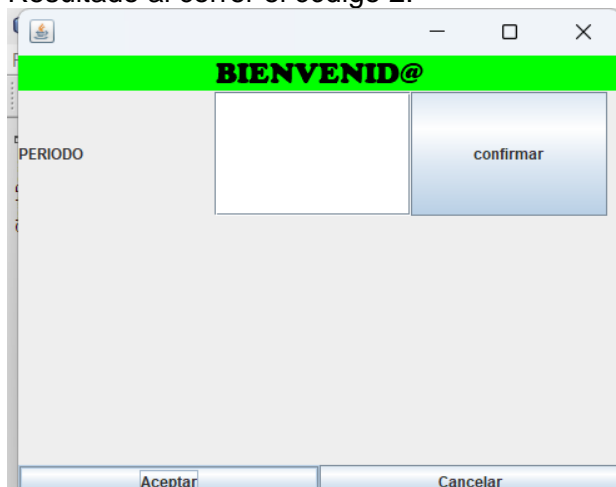


```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History [Icons]
142     });
143
144
145     } // Termina el Constructor
146
147
148     // Metodo
149     public void borrar() {
150         periodo.setText("");
151     }
152 }
153 public void llamar() {
154
155     String nada = periodo.getText();
156     if (nada.isEmpty()) {
157         JOptionPane.showMessageDialog(null, "Usted no ha ingresado ningun valor");
158     } else {
159         Autenticacionn3 terceraVentana = new Autenticacionn3();
160         terceraVentana.setVisible(true);
161         this.setVisible(false);
162     }
163 }
164
165
166
167
168 @Override
169 public void comprobar() {
170     String d = "";
171     if (d.equals(periodo.getText())) {
172         JOptionPane.showMessageDialog(null, "Usted no ha ingresado ningun valor");
173     } else {
174         confirmar.setVisible(false);
175
176         confirmar2.setVisible(true);
177
178         semestre.setVisible(true);
179         sem.setVisible(true);
180     }
181 }
182 }
183
184 @Override
185 public void visible() {
186     asignatura.setVisible(true);
187     materia.setVisible(true);
188     confirmar2.setVisible(false);
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x
Source History [Icons]
189     }
190
191
192
193     public static void main(String args[]){
194     Autenticacionn2 ventana= new Autenticacionn2 ();
195     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
196     } //cierra método
197
198     class Eventos implements ActionListener{
199     @Override //Esta sobre escrita por el metodo que ya tenia la madre
200     public void actionPerformed(ActionEvent ev){
201
202     if(ev.getSource()==aceptar){
203
204         llamar();
205     }
206     if(ev.getSource()==cancelar){
207     borrar(); // borrar es un metodo
208
209
210     }
211     if(ev.getSource()==confirmar){
212
213
214         comprobar();
215
216     }
217     if(ev.getSource()==confirmar2){
218         visible();
219
220
221     }
222
223
224
225     }
226     }
227
228
229     }
```

Resultado al correr el código 2:



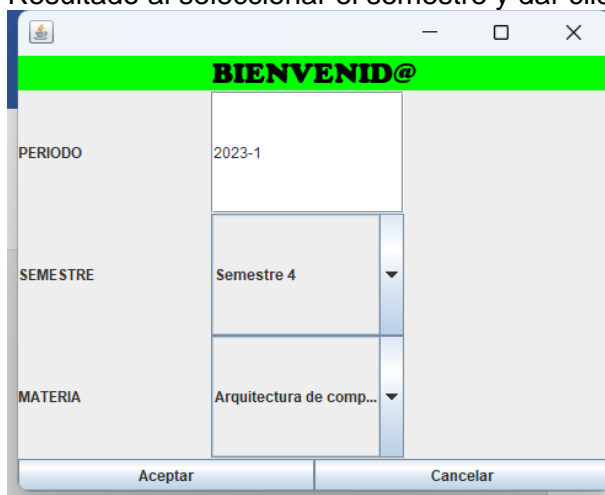
The screenshot shows a window titled "BIENVENID@" with a green header. Below the header, there are two input fields: "PERIODO" and "SEMESTRE". The "PERIODO" field is empty, and the "SEMESTRE" field is also empty. To the right of the "PERIODO" field is a blue button labeled "confirmar". At the bottom of the window, there are two buttons: "Aceptar" and "Cancelar".

Resultado al ingresar valores y dar clic en confirmar:



The screenshot shows the same window as before, but now the "PERIODO" field contains the text "2023-1" and the "SEMESTRE" field contains a dropdown menu with "Semestre 1" selected. The "confirmar" button is still present. The "Aceptar" and "Cancelar" buttons are at the bottom.

Resultado al seleccionar el semestre y dar clic nuevamente en confirmar:



The screenshot shows the same window as before, but now the "PERIODO" field contains "2023-1", the "SEMESTRE" field contains a dropdown menu with "Semestre 4" selected, and the "MATERIA" field contains a dropdown menu with "Arquitectura de comp..." selected. The "confirmar" button is still present. The "Aceptar" and "Cancelar" buttons are at the bottom.



Código de la ventana 3:

```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x
Source History
1 package autenticacionn;
2 import java.awt.Container;
3 import java.awt.BorderLayout;
4 import java.awt.Color;
5 import java.awt.Dimension;
6 import java.awt.Font;
7 import java.awt.GridLayout;
8 import java.awt.event.ActionEvent; //Acciones
9 import java.awt.event.ActionListener; //Detecta
10 import javax.swing.JFrame;
11 import javax.swing.JPanel; //Manteles
12 import javax.swing.JButton;
13 import javax.swing.JLabel;
14 import javax.swing.JTextField;
15 import javax.swing.JComboBox;
16 import javax.swing.JOptionPane;
17 import javax.swing.SwingConstants;
18 import java.util.ArrayList;
19 /**
20 *
21 * @author Jesus
22 */
23 public class Autenticacionn3 extends JFrame implements Validacion{
24     private JTextField calificacion1, calificacion2, calificacion3, calificacion4, calificacion5, calificacion6;
25     private JLabel titulo, cal1, cal2, cal3, cal4, cal5, cal6;
26     private JButton aceptar, cancelar, confirmar, confirmar2, salir;
27     private JPanel central, norte, sur;
28
29     public Autenticacionn3() {
30         Eventos manejador = new Eventos();
31         Container contenedor = getContentPane();
32         contenedor.setLayout(new BorderLayout());
33         central = new JPanel(new GridLayout(6, 2));
34         norte = new JPanel(new GridLayout(1, 1));
35         titulo = new JLabel("Ingrese las calificaciones de su materia seleccionada", SwingConstants.CENTER);
36         cal1 = new JLabel("Ingrese su calificacion de la unidad 1", SwingConstants.CENTER);
37         calificacion1 = new JTextField(40);
38         cal2 = new JLabel("Ingrese su calificacion de la unidad 2", SwingConstants.CENTER);
39
40         calificacion2 = new JTextField(40);
41
42         cal3 = new JLabel("Ingrese su calificacion de la unidad 3", SwingConstants.CENTER);
43
44         calificacion3 = new JTextField(40);
45
46         cal4 = new JLabel("Ingrese su calificacion de la unidad 4", SwingConstants.CENTER);

```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.j
Source History
47
48     calificacion4=new JTextField(40);
49
50     cal5= new JLabel ("Ingrese su calificacion de la unidad 5",SwingConstants.CENTER);
51
52     calificacion5=new JTextField(40);
53
54     cal6= new JLabel ("Ingrese su calificacion de la unidad 6",SwingConstants.CENTER);
55
56     calificacion6=new JTextField(40);
57     ;
58     sur= new JPanel(new GridLayout(1,1));
59     aceptar= new JButton("Guardar calificaciones");
60     aceptar.addActionListener(manejador);
61     cancelar= new JButton("Cancelar");
62     cancelar.addActionListener(manejador);
63
64
65
66     norte.add(titulo);
67     central.add(cal1);
68     central.add(calificacion1);
69
70     central.add(cal2);
71     central.add(calificacion2);
72
73     central.add(cal3);
74     central.add(calificacion3);
75     central.add(cal4);
76     central.add(calificacion4);
77     central.add(cal5);
78     central.add(calificacion5);
79     central.add(cal6);
80     central.add(calificacion6);
81     sur.add(aceptar);
82     sur.add(cancelar);
83
84     contenedor.add(norte,BorderLayout.NORTH);
85     contenedor.add(central,BorderLayout.CENTER);
86     contenedor.add(sur,BorderLayout.SOUTH);
87     titulo.setFont(new Font("Cooper Black",Font.PLAIN, 18));
88     titulo.setForeground(Color.black);
89     norte.setBackground(Color.green);
90     setSize(600,500); // pixeles, está es la mamá.
91     setVisible(true);
92 }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x
Source History [Icons]
93 public static void main(String args[]) {
94     Autenticacionn3 ventana = new Autenticacionn3();
95     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
96 }
97 public void borrar() {
98     calificacion1.setText("");
99     calificacion2.setText("");
100    calificacion3.setText("");
101    calificacion4.setText("");
102    calificacion5.setText("");
103    calificacion6.setText("");
104 }
105
106 @Override
107 public void comprobar() {
108
109     String c1 = calificacion1.getText();
110     int ca1 = Integer.parseInt(c1);
111     String c2 = calificacion2.getText();
112     int ca2 = Integer.parseInt(c2);
113     String c3 = calificacion3.getText();
114     int ca3 = Integer.parseInt(c3);
115     String c4 = calificacion4.getText();
116     int ca4 = Integer.parseInt(c4);
117     String c5 = calificacion5.getText();
118     int ca5 = Integer.parseInt(c5);
119     String c6 = calificacion6.getText();
120     int ca6 = Integer.parseInt(c6);
121
122
123     if((ca1<69) ){
124         calificacion1.setText("N/A");
125
126
127     }if((ca2<69) ){
128         calificacion2.setText("N/A");
129
130
131     }if((ca3<69) ){
132         calificacion3.setText("N/A");
133
134
135     }if((ca4<69) ){
136         calificacion4.setText("N/A");
137
138 }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x
Source History
139         }if((ca5<69) ){
140             calificacion5.setText("N/A");
141         }if((ca6<69) ){
142             calificacion6.setText("N/A");
143         }if((cal>100) ||(ca2>100) ||(ca3>100) || (ca4>100) || (ca5>100) ||(ca6>100)){
144             JOptionPane.showMessageDialog(null,"Usted ha ingresado valores mayores a 100 ");
145         }
146     }
147     @Override
148     public void visible(){
149         String n1=calificacion1.getText();
150         String n2=calificacion2.getText();
151         String n3=calificacion3.getText();
152         String n4=calificacion4.getText();
153         String n5=calificacion5.getText();
154         String n6=calificacion6.getText();
155         if(n1.isEmpty() ||n2.isEmpty() ||n3.isEmpty() ||n4.isEmpty() ||n5.isEmpty() ||n6.isEmpty()){
156             JOptionPane.showMessageDialog(null,"Usted no ha ingresado todos los valores ");
157         }
158     }
159
160
161
162     class Eventos implements ActionListener{
163
164         @Override
165         public void actionPerformed(ActionEvent ev) {
166             if(ev.getSource()==cancelar){
167                 borrar();
168             }
169             if(ev.getSource()==aceptar){
170
171                 visible();
172                 comprobar();
173                 JOptionPane.showMessageDialog(null,"Sus calificaciones han sido guardadas correctamente ");
174                 System.exit(0);
175             }
176
177         }
178
179     }
180
181 }
```



Resultado al correr el código 3:

La imagen muestra una ventana de software con un título que dice "Ingrese las calificaciones de su materia seleccionada". La ventana tiene un fondo gris y una barra de título azul. En el interior, hay una lista de unidades con sus respectivas calificaciones:

| Unidad | Calificación |
|--|--------------|
| Ingrese su calificación de la unidad 1 | |
| Ingrese su calificación de la unidad 2 | |
| Ingrese su calificación de la unidad 3 | |
| Ingrese su calificación de la unidad 4 | |
| Ingrese su calificación de la unidad 5 | |
| Ingrese su calificación de la unidad 6 | |

En la parte inferior de la ventana, hay dos botones: "Guardar calificaciones" y "Cancelar".

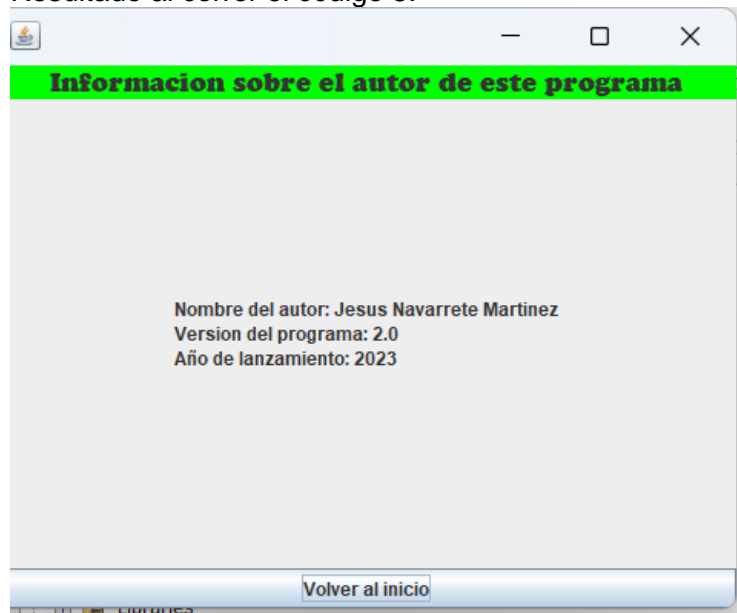
Código de la ventana 4:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History [Icons]
6 package autenticacionn;
7 import java.awt.Container;
8 import java.awt.BorderLayout;
9 import java.awt.Color;
10 import java.awt.Font;
11 import java.awt.GridLayout;
12 import java.awt.event.ActionEvent;//Acciones
13 import java.awt.event.ActionListener;//Detecta
14 import javax.swing.JFrame;
15 import javax.swing.JPanel;
16 import javax.swing.JButton;
17 import javax.swing.JLabel;
18 import javax.swing.JTextField;
19 import javax.swing.JComboBox;
20 import javax.swing.SwingConstants;
21 public class Autor extends JFrame {
22     private JLabel titulo,informacion;
23
24     private JButton volver;
25     private JPanel central,norte,sur;
26
27     public Autor() {
28         Eventos manejador=new Eventos();
29         Container contenedor = getContentPane();
30         contenedor.setLayout(new BorderLayout());
31         central = new JPanel(new GridLayout(1,1));
32         norte = new JPanel (new GridLayout(1,1));
33         sur= new JPanel(new GridLayout(1,1));
34         titulo = new JLabel ("Informacion sobre el autor de este programa", SwingConstants.CENTER);
35         informacion = new JLabel ("<html>Nombre del autor: Jesus Navarrete Martinez<br/>"
36             + " Version del programa: 2.0<br/>Año de lanzamiento: 2023", SwingConstants.CENTER);
37         volver=new JButton ("Volver al inicio");
38         volver.addActionListener(manejador);
39         norte.add(titulo);
40         central.add(informacion);
41         sur.add(volver);
42         contenedor.add(norte, BorderLayout.NORTH);
43         contenedor.add(central, BorderLayout.CENTER);
44         contenedor.add(sur, BorderLayout.SOUTH);
45         titulo.setFont(new Font ("Cooper Black",Font.PLAIN, 18));
46         norte.setBackground(Color.green);
47         setSize(500,400); // pixeles, está es la mamá.
48         setVisible(true);
49     }
50 }
```



```
50 public static void main(String args[]){
51     Autor ventana = new Autor();
52     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
53 }
54 public void traer(){
55     Autenticacionn terceraVentana= new Autenticacionn();
56     terceraVentana.setVisible(true);
57     this.setVisible(false);
58 }
59 class Eventos implements ActionListener{
60     @Override
61     public void actionPerformed(ActionEvent ev) {
62         if (ev.getSource() == volver) {
63             traer();
64         }
65     }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
```

Resultado al correr el código 5:





Código de la ventana 5:

```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
6 package autenticacionn;
7 import java.awt.Container;
8 import java.awt.BorderLayout;
9 import java.awt.Color;
10 import java.awt.Font;
11 import java.awt.GridLayout;
12 import java.awt.event.ActionEvent; //Acciones
13 import java.awt.event.ActionListener; //Detecta
14 import javax.swing.JFrame;
15 import javax.swing.JPanel; //Manteles
16 import javax.swing.JButton;
17 import javax.swing.JLabel;
18 import javax.swing.JTextField;
19 import javax.swing.JComboBox;
20 import javax.swing.SwingConstants;
21 public class Instructivo extends JFrame {
22     private JLabel titulo, informacion;
23
24     private JButton volver;
25     private JPanel central, norte, sur;
26
27     public Instructivo() {
28         Eventos manejador = new Eventos();
29         Container contenedor = getContentPane();
30         contenedor.setLayout(new BorderLayout());
31         central = new JPanel(new GridLayout(1,1));
32         norte = new JPanel(new GridLayout(1,1));
33         sur = new JPanel(new GridLayout(1,1));
34         titulo = new JLabel("Bienvenido a nuestra guía de usuario", SwingConstants.CENTER);
35         informacion = new JLabel("<html>Instructivo ventana 1: <br/>"
36             + " paso 1: Escriba su usuario y contraseña en los apartados correspondientes.<br/>"
37             + " paso 2: Dar clic en el boton aceptar, el cual verificara si el usuario y"
38             + " contraseña ingresados son correctos de ser asi podra avanzar a la siguiente ventana.<br/>"
39             + " paso3: Con el boton de cancelar podra borrar su usuario y contraseña ingresados"
40             + "<br/>Nota: si el usuario o la contraseña son incorrectos no podra avanzar a la"
41             + " siguiente ventana, solo tendra 3 oportunidades para intentar ingresar<br/>"
42             + "una vez agotadas sus oportunidades el programa se cerrara automaticamente<br/>"
43             + "<br/>"
44             + "Instructivo Ventana 2:<br/>"
45             + "pasol: En esta ventana usted podra escribir el periodo al que pertenece y posteriormete"
46             + " podra seleccionar el semestre que esta cursando al hacer esto ,automaticamente "
47             + "aparecera la materia que debe cursarse de dicho semestre.<br/>"
48             + "paso 2: una vez seleccionado el semestre y materia usted debe dar"
49             + " click en el boton aceptar para avanzar a la siguiente ventana.<br/>"
50             + "Nota: con el boton cancelar usted podra volver a escribir el"
51             + " periodo al que pertenece asi elegir otro semestre y materia.<br/>"

```



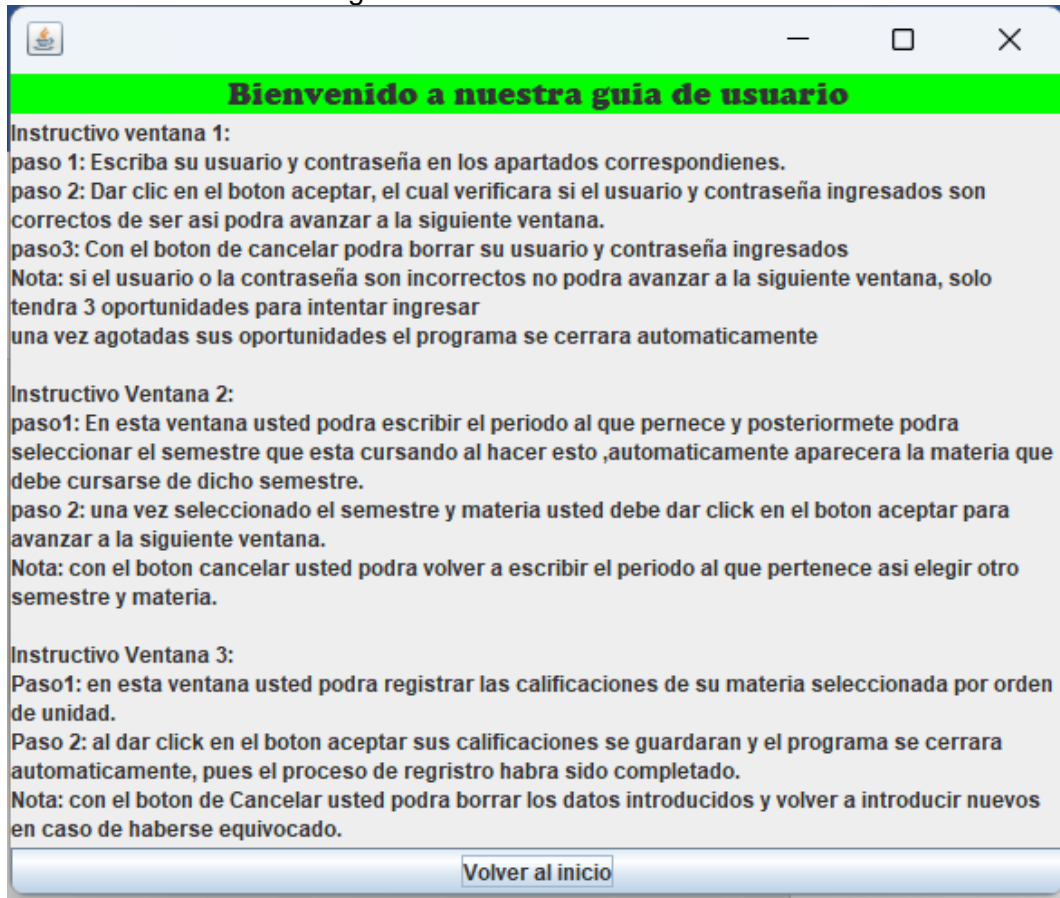

```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
52 + "<br/>Instructivo Ventana 3:<br/>"
53 + "Pasol: en esta ventana usted podra registrar las calificaciones"
54 + " de su materia seleccionada por orden de unidad.<br/>"
55 + "Paso 2: al dar click en el boton aceptar sus calificaciones se guardaran"
56 + " y el programa se cerrara automaticamente, pues el proceso de registro"
57 + " habra sido completado.<br/>"
58 + "Nota: con el boton de Cancelar usted podra borrar los datos introducidos y"
59 + " volver a introducir nuevos en caso de haberse equivocado.", SwingConstants.CENTER);
60 volver=new JButton ("Volver al inicio");
61 volver.addActionListener(manejador);
62 norte.add(titulo);
63 central.add(informacion);
64 sur.add(volver);
65 contenedor.add(norte, BorderLayout.NORTH);
66 contenedor.add(central, BorderLayout.CENTER);
67 contenedor.add(sur, BorderLayout.SOUTH);
68 titulo.setFont(new Font("Cooper Black", Font.PLAIN, 18));
69 norte.setBackground(Color.green);
70 setSize(600,500); // pixeles, está es la mamá.
71 setVisible(true);
72 }
73 public static void main(String args[]){
74     Autor ventana = new Autor();
75     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
76 }
77 public void traer(){
78     Autenticacionn terceraVentana= new Autenticacionn();
79     terceraVentana.setVisible(true);
80     this.setVisible(false);
81 }
82 class Eventos implements ActionListener{
83     @Override
84     public void actionPerformed(ActionEvent ev) {
85         if(ev.getSource()==volver){
86             traer();
87         }
88     }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }

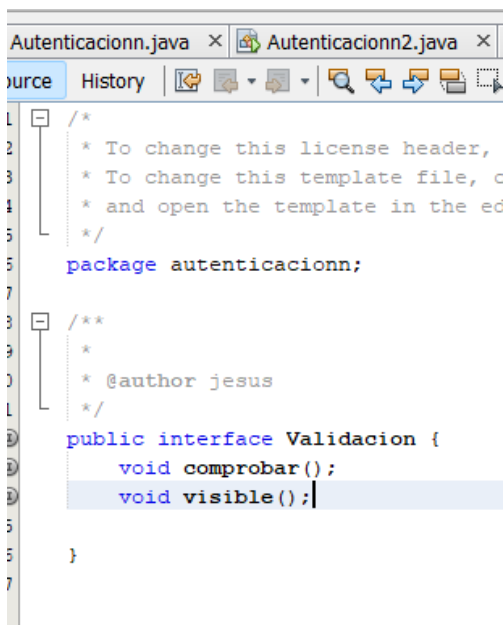
```



Resultado al correr el código 5:



Código de la interfaz y los métodos utilizados:



V. Conclusiones:

El desarrollo de una interfaz gráfica nos sirve para crear un programa o aplicación más interactivo y llamativo para el usuario pues en ellas se emplean algunos objetos visuales como ventanas, botones, paneles e inclusive podemos hacer uso de colores para personalizarlo a nuestro gusto o según las necesidades que existan.

Las interfaces que podemos llegar a crear pueden ser muy atractivas y funcionales, ya que permiten al usuario realizar alguna tarea de forma más sencilla y eficiente para el usuario. Con el desarrollo de este proyecto pude darme cuenta de que cada parte del código para la creación de la interfaz es de esencial importancia para su correcto funcionamiento, pues un ligero desacomodo en el código puede cambiar por completo el aspecto y la funcionalidad de la ventana que intentamos crear, esta se puede personalizar como el tipo de letra, color, tamaño, posición, etc. según las necesidades del usuario y sus necesidades.

A pesar de todo esto no podemos dejar de lado el uso de la herencia simple en el desarrollo de este programa, pues con ella pude organizar mejor mi código y reutilizar algunas variables ya declaradas en otras clases sin tener que volver a estructurar todo nuevamente, con ello se puede utilizar una jerarquía de sus funcionalidades, los modificadores de acceso también juegan un papel importante en el desarrollo de programas cuando se utiliza la herencia ya que de ellos depende si algunos elementos de las clases derivadas pueden utilizar métodos o atributos de la clase base para realizar funcionalidades y tomar valores según las necesidades de la situación.

Durante el desarrollo e implementación de una interfaz para utilizar métodos abstractos en otras clases pude darme cuenta que las interfaces nos permiten crear un conjunto de métodos específicos, los cuales pueden llegar a ser utilizados o implementados en cualquier otra clase, pudiendo realizar funciones diferentes según los requerimientos de la situación, las interfaces permiten la reutilización de código y la implementación del polimorfismo a la vez lo que puede hacer a nuestro código más flexible y estructurado, pues los métodos creados en la interfaz no se les agrega cuerpo o funcionalidades, sin embargo al implementarlos en otra clase si se les agrega cuerpo y funciones dentro del método además de que la misma cantidad de métodos declarados en la interfaz deben ser también implementados en la clase deseada para evitar errores, sin excluir ni uno solo.

