

Nombre de la práctica	UNIDAD 6. Flujos y Archivos			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 | [Icons]
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 ENCRYPTAR", SwingConstants.CENTER);
35     entradaLbl = new JLabel("Texto a Encriptar", SwingConstants.CENTER);
36     salida = new JLabel("ENCRYPTAR", SwingConstants.CENTER);
37     salidaCadena = new JLabel ("DESENCRIPTAR", SwingConstants.CENTER);
38     encriptar = new JButton("ENCRYPTAR");
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     descriptado=String.valueOf(array);
106     salidaCadena.setText(descriptado);
107 }//cierra metodo
108
109 public void borrar(){
110     entrada.setText("");
111     salida.setText("");
112     salidaCadena.setText("");
113 }
114
115
116
117 class Eventos implements ActionListener{
118     @Override
119     public void actionPerformed(ActionEvent ev){
120         if(ev.getSource()==encriptar){
121             activar();
122         }
123
124         if(ev.getSource()==desencriptar){
125             desactivar();
126         }
127
128         if(ev.getSource()==limpiar){
129             borrar();
130         }
131
132         if(ev.getSource()==salir){
133             System.exit(0);
134         }
135     }
136 }
137
138
139
140 }
```

Resultado de correr el código Encriptador :



EJERCICIO PARA ENCRYPTAR	
Texto a Encriptar	
ENCRYPTAR	ENCRYPTAR
DESENCRIPTAR	DESENCRIPTAR
Limpiar	Salir



Código del proyecto Interfaz Gráfica:

Código de la ventana 1:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java
Source History
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
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("?");
```



```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
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);
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 });

```



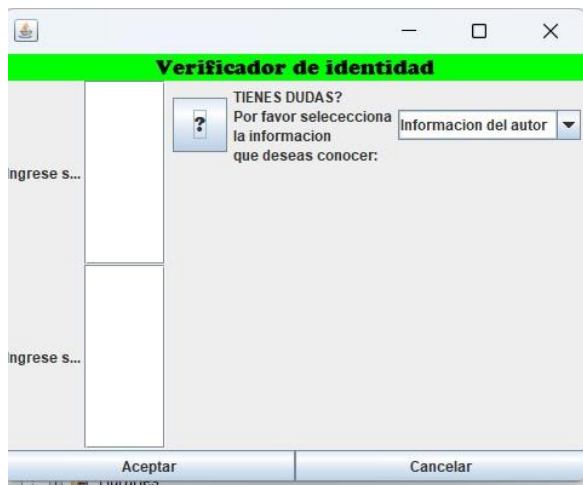
```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.ja
Source History
95 } // termina el constructor.
96 public static void main(String args[]){
97     Autenticacionn ventana = new Autenticacionn();
98     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
99 }
100 public void borrar(){
101     entrada.setText("");
102     contra.setText("");
103 }
104 public void traer(){
105     Autenticacionn ven= new Autenticacionn();
106     ven.setVisible(true);
107     this.setVisible(false);
108 }
109 public class InvalidUserException extends Exception {
110 public InvalidUserException(String message) {
111     super(message);
112 }
113 }
114
115 public void verificarUsuario(){
116
117     try{
118
119         if(!user.equals(entrada.getText()) || !clave.equals(contra.getText())){
120             contador++;
121
122             entrada.setText("");
123             contra.setText("");
124
125             if(contador==3){
126                 System.exit(0);
127             }
128
129             throw new InvalidUserException("Usuario incorrecto");
130
131         }
132
133         Autenticacionn2 segundaVentana= new Autenticacionn2();
134         segundaVentana.setVisible(true);
135         this.setVisible(false);
136
137
138     }catch(InvalidUserException e){
139         JOptionPane.showMessageDialog(null,"Contraseña o usuario incorrecto");
140     }
141 }
```




```
142 |
143 |     }
144 |
145 |     class Eventos implements ActionListener{
146 |         @Override
147 |         public void actionPerformed(ActionEvent ev) {
148 |             if(ev.getSource() == cancelar) {
149 |                 borrar();
150 |             }
151 |             if(ev.getSource() == aceptar) {
152 |
153 |                 verificarUsuario();
154 |             }
155 |             if(ev.getSource() == ayuda) {
156 |                 s.setVisible(true);
157 |                 regresar.setVisible(true);
158 |                 info.setVisible(true);
159 |                 info.setPopupVisible(true);
160 |                 contadorc++;
161 |                 if(contadorc == 2) {
162 |                     traer();
163 |                 }
164 |             }
165 |         }
166 |     }
167 | }
168 |
169 | }
170 |
```

Resultado al correr el código 1:

Resultado al accionar el botón “?”:



Verificador de identidad

ingrese s...

ingrese s...

TIENES DUDAS?
Por favor seleccione la informacion que deseas conocer:

Informacion del autor

Aceptar Cancelar



Código de la ventana 2:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacionn.java
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 Instructivo.j
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     aceptar.setVisible(false);
61     cancelar=new JButton("Cancelar");
62     cancelar.addActionListener(manejador);
63     confirmar= new JButton("confirmar");
64     confirmar.addActionListener(manejador);
65     confirmar.setPreferredSize(new Dimension(45, 45));
66     confirmar2= new JButton("confirmar");
67     confirmar2.addActionListener(manejador);
68     confirmar2.setPreferredSize(new Dimension(45, 45));
69     confirmar2.setVisible(false);
70     inicio.add(texto);
71     run.add(confirmar);
72     central.add(ciclo);
73     central.add(periodo);
74     central.add(confirmar);
75     central.add(semestre);
76     central.add(sem);
77     central.add(confirmar2);
78     central.add(asignatura);
79     central.add(materia);
80     ultimo.add(aceptar);
81     ultimo.add(cancelar);
82     contenedor.add(run, BorderLayout.EAST);
83     contenedor.add(central, BorderLayout.CENTER);
84     contenedor.add(inicio, BorderLayout.NORTH);
85     contenedor.add(ultimo, BorderLayout.SOUTH);
86     setSize(500, 400);
87     setVisible(true);
88     texto.setForeground(Color.BLACK);
89     texto.setFont(new Font("Cooper Black", Font.PLAIN, 24));
90     sem.addActionListener(e -> {
91         String opcionSeleccionada = (String) sem.getSelectedItem();
92         if (opcionSeleccionada.equals("Semestre 1")) {
93
94             String[] subopciones1 = {"Fundamentos de programacion"};

```



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

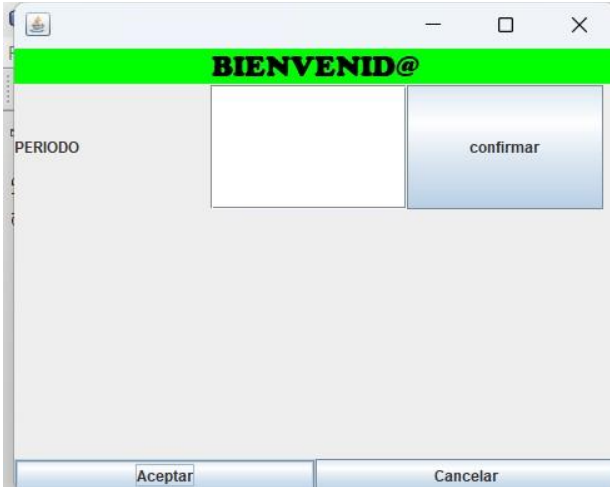



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x E
Source History
142     }
143     });
144
145
146     //Termina el Constructor
147
148
149     //Metodo
150     public void borrar() {
151         periodo.setText("");
152     }
153 }
154 public void llamar() {
155
156     Autenticacionn3 terceraVentana= new Autenticacionn3();
157     terceraVentana.setVisible(true);
158     this.setVisible(false);
159
160 }
161
162
163
164
165
166 @Override
167 public void comprobar() {
168
169     try{
170         String periodo2= periodo.getText();
171         if(periodo2.isEmpty()){
172             throw new NullPointerException("El periodo ingresado está vacío");
173         }
174         if(periodo2.matches("\\d{4}-\\d{2}")){
175
176             confirmar.setVisible(false);
177             confirmar2.setVisible(true);
178             semestre.setVisible(true);
179             sem.setVisible(true);
180         }else{
181             JOptionPane.showMessageDialog(null,"Usted no ha ingresado un periodo valido");
182         }
183     }catch(NullPointerException d){
184         JOptionPane.showMessageDialog(null,"El periodo no puede estar vacío por favor ingresa tu periodo");
185     }
186 }
187
188
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Auto
Source History
189
190
191 }
192
193 @Override
194 public void visible() {
195     asignatura.setVisible(true);
196     materia.setVisible(true);
197     confirmar2.setVisible(false);
198     aceptar.setVisible(true);
199 }
200
201
202
203 public static void main(String args[]) {
204     Autenticacionn2 ventana = new Autenticacionn2 ();
205     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
206 } //cierra método
207
208 class Eventos implements ActionListener {
209     @Override
210     public void actionPerformed(ActionEvent ev) {
211
212         if (ev.getSource() == aceptar) {
213
214             llamar();
215         }
216         if (ev.getSource() == cancelar) {
217             borrar();
218         }
219         if (ev.getSource() == confirmar) {
220
221             comprobar();
222
223         }
224         if (ev.getSource() == confirmar2) {
225             visible();
226
227         }
228
229     }
230
231
232
233 }
234
235 }
```

Resultado al correr el código 2:



Resultado al ingresar valores y dar clic en confirmar:



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





Código de la ventana 3:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.
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, promedio;
27     private JPanel central, norte, sur;
28     int contador=0;
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         calificacion2=new JTextField(40);
40         cal3= new JLabel ("Ingrese su calificacion de la unidad 3",SwingConstants.CENTER);
41         calificacion3=new JTextField(40);
42         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.java x
Source History
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 promedio=new JButton("Calcular Promedio");
64 promedio.addActionListener(manejador);
65
66
67 norte.add(titulo);
68 central.add(cal1);
69 central.add(calificacion1);
70
71 central.add(cal2);
72 central.add(calificacion2);
73
74 central.add(cal3);
75 central.add(calificacion3);
76 central.add(cal4);
77 central.add(calificacion4);
78 central.add(cal5);
79 central.add(calificacion5);
80 central.add(cal6);
81 central.add(calificacion6);
82 sur.add(aceptar);
83 sur.add(cancelar);
84 sur.add(promedio);
85
86
87 contenedor.add(norte, BorderLayout.NORTH);
88 contenedor.add(central, BorderLayout.CENTER);
89 contenedor.add(sur, BorderLayout.SOUTH);
90 titulo.setFont(new Font("Cooper Black",Font.PLAIN, 18));
91 titulo.setForeground(Color.black);
92 norte.setBackground(Color.green);
93 setSize(600,500); // pixeles, está es la mamá.
94 setVisible(true);
```




```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x
Source History [Icons]
95 }
96 public static void main(String args[]) {
97     Autenticacionn3 ventana = new Autenticacionn3();
98     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
99 }
100 public void borrar() {
101     calificacion1.setText("");
102     calificacion2.setText("");
103     calificacion3.setText("");
104     calificacion4.setText("");
105     calificacion5.setText("");
106     calificacion6.setText("");
107 }
108
109 @Override
110 public void comprobar() {
111     try{
112         String c1 = calificacion1.getText();
113         int ca1 = Integer.parseInt(c1);
114         String c2 = calificacion2.getText();
115         int ca2 = Integer.parseInt(c2);
116         String c3 = calificacion3.getText();
117         int ca3 = Integer.parseInt(c3);
118         String c4 = calificacion4.getText();
119         int ca4 = Integer.parseInt(c4);
120         String c5 = calificacion5.getText();
121         int ca5 = Integer.parseInt(c5);
122         String c6 = calificacion6.getText();
123         int ca6 = Integer.parseInt(c6);
124
125         if((ca1 < 69 && ca1 > 0) ){
126             calificacion1.setText("N/A");
127
128         }if((ca2 < 69 && ca2 > 0) ){
129             calificacion2.setText("N/A");
130
131         }if((ca3 < 69 && ca3 > 0) ){
132             calificacion3.setText("N/A");
133
134         }if((ca4 < 69 && ca4 > 0) ){
135             calificacion4.setText("N/A");
136
137         }if((ca5 < 69 && ca5 > 0) ){
138             calificacion5.setText("N/A");
139
140         }if((ca6 < 69 && ca6 > 0) ){
141             calificacion6.setText("N/A");
142         }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java
Source History
142         }if((ca5<69 && ca5>0) ){
143             calificacion5.setText("N/A");
144         }if((ca6<69 && ca6>0) ){
145             calificacion6.setText("N/A");
146         }
147
148
149
150
151     }catch (NumberFormatException e){
152         JOptionPane.showMessageDialog(null,"Usted no ha ingresado valores numericos ");
153     }
154 }
155
156 @Override
157 public void visible(){
158     try{
159         String c1 = calificacion1.getText();
160         int ca1 = Integer.parseInt(c1);
161         String c2= calificacion2.getText();
162         int ca2= Integer.parseInt(c2);
163         String c3= calificacion3.getText();
164         int ca3= Integer.parseInt(c3);
165         String c4= calificacion4.getText();
166         int ca4= Integer.parseInt(c4);
167         String c5= calificacion5.getText();
168         int ca5= Integer.parseInt(c5);
169         String c6= calificacion6.getText();
170         int ca6= Integer.parseInt(c6);
171         if((ca1>100) || (ca2>100) || (ca3>100) || (ca4>100) || (ca5>100) || (ca6>100)){
172             JOptionPane.showMessageDialog(null,"Usted ha ingresado valores mayores a 100 ");
173         }
174         if ((ca1<0) || (ca2<0) || (ca3<0) || (ca4<0) || (ca5<0) || (ca6<0)) {
175             throw new IllegalArgumentException("El parámetro debe ser un valor positivo");
176         }
177     }
178 }
179
180
181     }catch(IllegalArgumentException b){
182         JOptionPane.showMessageDialog(null,"Error: Los valores ingresados no pueden ser menores a cero ");
183     }
184 }
185 public void calcular(){
186     try{
187         String c1 = calificacion1.getText();
188         int ca1 = Integer.parseInt(c1);
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x Exce
Source History
188 int ca1 = Integer.parseInt(c1);
189 String c2= calificacion2.getText();
190 int ca2= Integer.parseInt(c2);
191 String c3= calificacion3.getText();
192 int ca3= Integer.parseInt(c3);
193 String c4= calificacion4.getText();
194 int ca4= Integer.parseInt(c4);
195 String c5= calificacion5.getText();
196 int ca5= Integer.parseInt(c5);
197 String c6= calificacion6.getText();
198 int ca6= Integer.parseInt(c6);
199
200
201
202 int divisor;
203 divisor= Integer.parseInt(JOptionPane.showInputDialog("Introduce la cantidades de unidades a promediar"));
204 int promedioFinal=(ca1+ca2+ca3+ca4+ca5+ca6)/divisor;
205 JOptionPane.showMessageDialog(null,"Su promedio final es: "+promedioFinal);
206 }catch(ArithmeticException e){
207     JOptionPane.showMessageDialog(null,"Error: las calificaciones ingresadas no pueden ser divididas entre 0 ");
208 }
209
210
211
212
213 class Eventos implements ActionListener{
214
215     @Override
216     public void actionPerformed(ActionEvent ev) {
217         if(ev.getSource()==cancelar){
218             borrar();
219         }
220         if(ev.getSource()==aceptar){
221
222             contador++;
223             if(contador==1){
224                 visible();
225                 comprobar();
226             }
227
228             if(contador==2){
229
230                 JOptionPane.showMessageDialog(null,"Sus calificaciones se guardaron con exito ");
231                 System.exit(0);
232             }
233         }
234
235         if(ev.getSource()==promedio){
236             calcular();
237         }
238     }
239 }
240
241
242
243
244
245
246
247
248
249
250
251
```



Resultado al correr el código 3:

La imagen muestra una ventana de aplicación con un título que dice "Ingrese las calificaciones de su materia seleccionada". El título tiene un fondo verde y el texto es blanco. Dentro de la ventana, hay una lista de seis unidades, cada una con un campo de entrada de texto a su derecha:

- Ingrese su calificacion de la unidad 1
- Ingrese su calificacion de la unidad 2
- Ingrese su calificacion de la unidad 3
- Ingrese su calificacion de la unidad 4
- Ingrese su calificacion de la unidad 5
- Ingrese su calificacion de la unidad 6

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



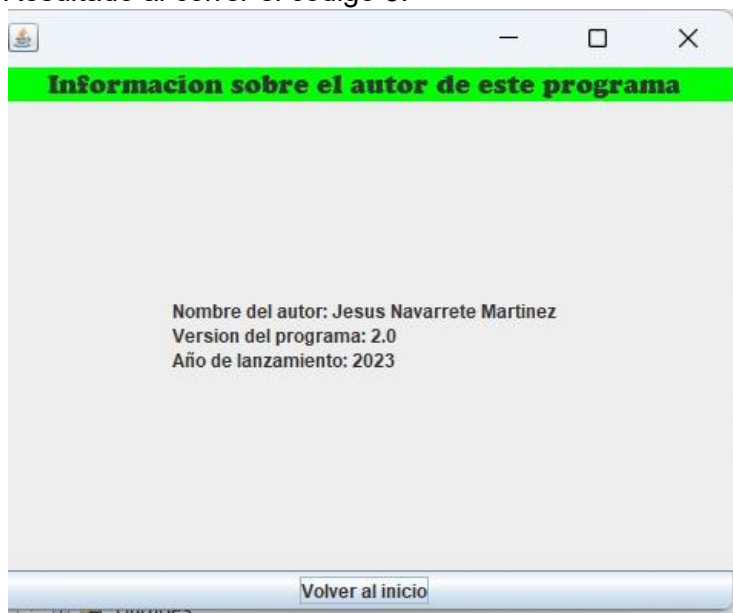
Código de la ventana 4:

```
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;
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 guia 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/>"

```



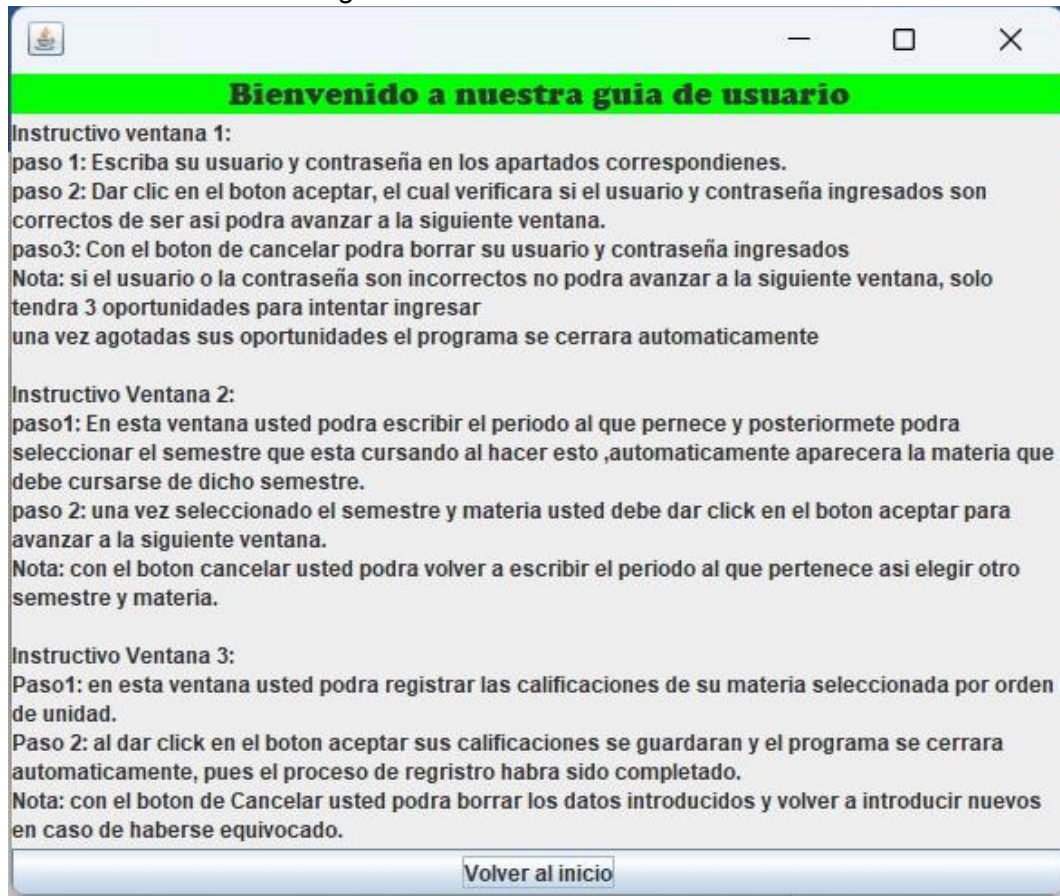
```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
52 + "<br/>Instructivo Ventana 3:<br/>"
53 + "Paso1: 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:

```
Autenticacionn.java x Autenticacionn2.java x
Source History
1  /*
2   * To change this license header,
3   * To change this template file, c
4   * and open the template in the ed
5   */
6  package autenticacionn;
7
8  /**
9   *
10  * @author jesus
11  */
12  public interface Validacion {
13      void comprobar();
14      void visible();
15  }
16
17  }
```



Unidad 5 Excepciones

Código de la ventana 1:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java
Source History
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
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("?");
```




```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History [Icons]
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);
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 });

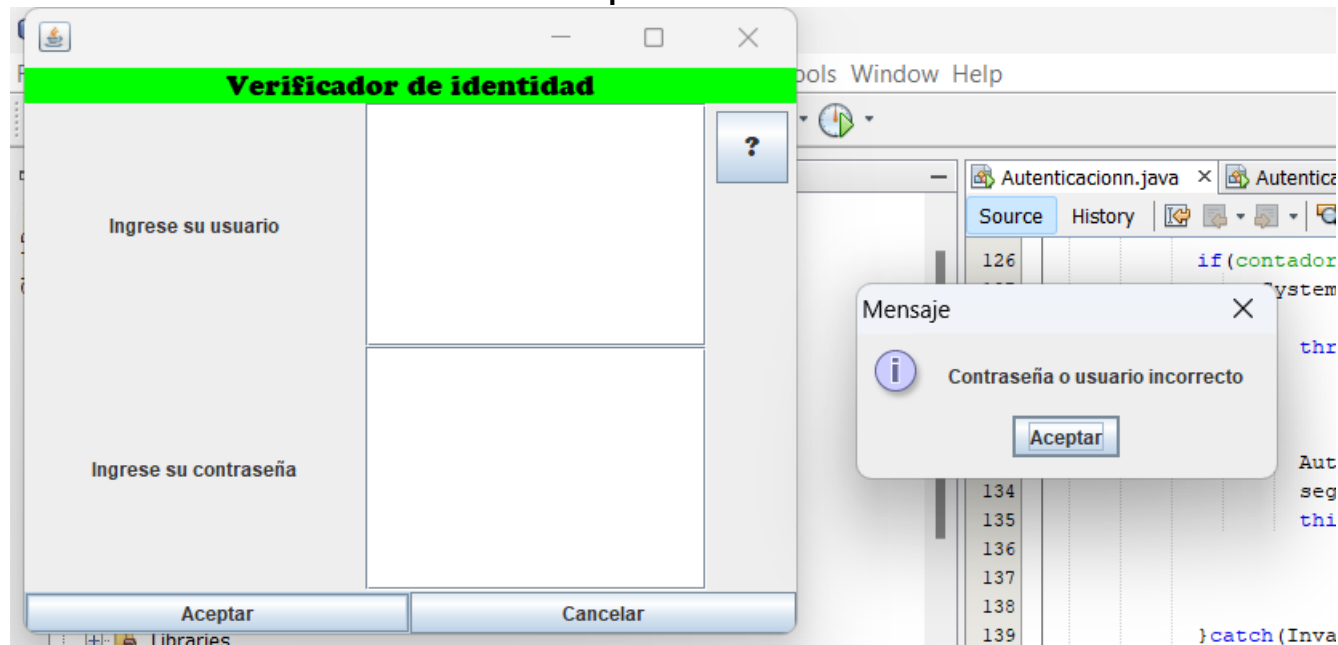
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.ja
Source History
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 }
104 public void traer(){
105     Autenticacionn ven= new Autenticacionn();
106     ven.setVisible(true);
107     this.setVisible(false);
108 }
109 public class InvalidUserException extends Exception {
110     public InvalidUserException(String message) {
111         super(message);
112     }
113 }
114 }
115 public void verificarUsuario(){
116     try{
117         if(!user.equals(entrada.getText()) || !clave.equals(contra.getText())){
118             contador++;
119             entrada.setText("");
120             contra.setText("");
121             if(contador==3){
122                 System.exit(0);
123             }
124             throw new InvalidUserException("Usuario incorrecto");
125         }
126         Autenticacionn2 segundaVentana= new Autenticacionn2();
127         segundaVentana.setVisible(true);
128         this.setVisible(false);
129     }catch(InvalidUserException e){
130         JOptionPane.showMessageDialog(null,"Contraseña o usuario incorrecto");
131     }
132 }
```

En este apartado realice una excepción personalizada (InvalidUserException) la cual tiene como función comparar si el usuario y contraseña son correctos o erróneos en caso de ser erróneos se la excepción lanza un mensaje indicando que los datos ingresados son incorrectos.

Resultado cuando entra en función la excepción:





Código de la ventana 2:

```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validac
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 Instructivo.j
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     aceptar.setVisible(false);
61     cancelar=new JButton ("Cancelar");
62     cancelar.addActionListener(manejador);
63     confirmar= new JButton("confirmar");
64     confirmar.addActionListener(manejador);
65     confirmar.setPreferredSize(new Dimension(45, 45));
66     confirmar2= new JButton("confirmar");
67     confirmar2.addActionListener(manejador);
68     confirmar2.setPreferredSize(new Dimension(45, 45));
69     confirmar2.setVisible(false);
70     inicio.add(texto);
71     run.add(confirmar);
72     central.add(ciclo);
73     central.add(periodo);
74     central.add(confirmar);
75     central.add(semestre);
76     central.add(sem);
77     central.add(confirmar2);
78     central.add(asignatura);
79     central.add(materia);
80     ultimo.add(aceptar);
81     ultimo.add(cancelar);
82     contenedor.add(run, BorderLayout.EAST);
83     contenedor.add(central, BorderLayout.CENTER);
84     contenedor.add(inicio, BorderLayout.NORTH);
85     contenedor.add(ultimo, BorderLayout.SOUTH);
86     setSize(500,400);
87     setVisible(true);
88     texto.setForeground(Color.BLACK);
89     texto.setFont(new Font ("Cooper Black", Font.PLAIN,24));
90     sem.addActionListener(e -> {
91         String opcionSeleccionada = (String) sem.getSelectedItem();
92         if (opcionSeleccionada.equals("Semestre 1")) {
93
94             String[] subopciones1 = {"Fundamentos de programacion"};

```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instruc
Source History
95 materia.removeAllItems();
96 for (String subopcion : subopciones1) {
97     materia.addItem(subopcion);
98 }
99 } else if (opcionSeleccionada.equals("Semestre 2")) {
100     String[] subopciones2 = {"Programacion orientada a"};
101     materia.removeAllItems();
102     for (String subopcion : subopciones2) {
103         materia.addItem(subopcion);
104     }
105 } else if (opcionSeleccionada.equals("Semestre 3")) {
106     String[] subopciones2 = {"Sistemas Operativos"};
107     materia.removeAllItems();
108     for (String subopcion : subopciones2) {
109         materia.addItem(subopcion);
110     }
111 } else if (opcionSeleccionada.equals("Semestre 4")) {
112     String[] subopciones2 = {"Arquitectura de computadoras"};
113     materia.removeAllItems();
114     for (String subopcion : subopciones2) {
115         materia.addItem(subopcion);
116     }
117 } else if (opcionSeleccionada.equals("Semestre 5")) {
118     String[] subopciones2 = {"Redes de computadoras"};
119     materia.removeAllItems();
120     for (String subopcion : subopciones2) {
121         materia.addItem(subopcion);
122     }
123 } else if (opcionSeleccionada.equals("Semestre 6")) {
124     String[] subopciones2 = {"Programacion web"};
125     materia.removeAllItems();
126     for (String subopcion : subopciones2) {
127         materia.addItem(subopcion);
128     }
129 } else if (opcionSeleccionada.equals("Semestre 7")) {
130     String[] subopciones2 = {"Internet de las cosas"};
131     materia.removeAllItems();
132     for (String subopcion : subopciones2) {
133         materia.addItem(subopcion);
134     }
135 }
136 else {
137     String[] subopciones3 = {"Inteligencia artificial"};
138     materia.removeAllItems();
139     for (String subopcion : subopciones3) {
140         materia.addItem(subopcion);
141     }
142 }
```

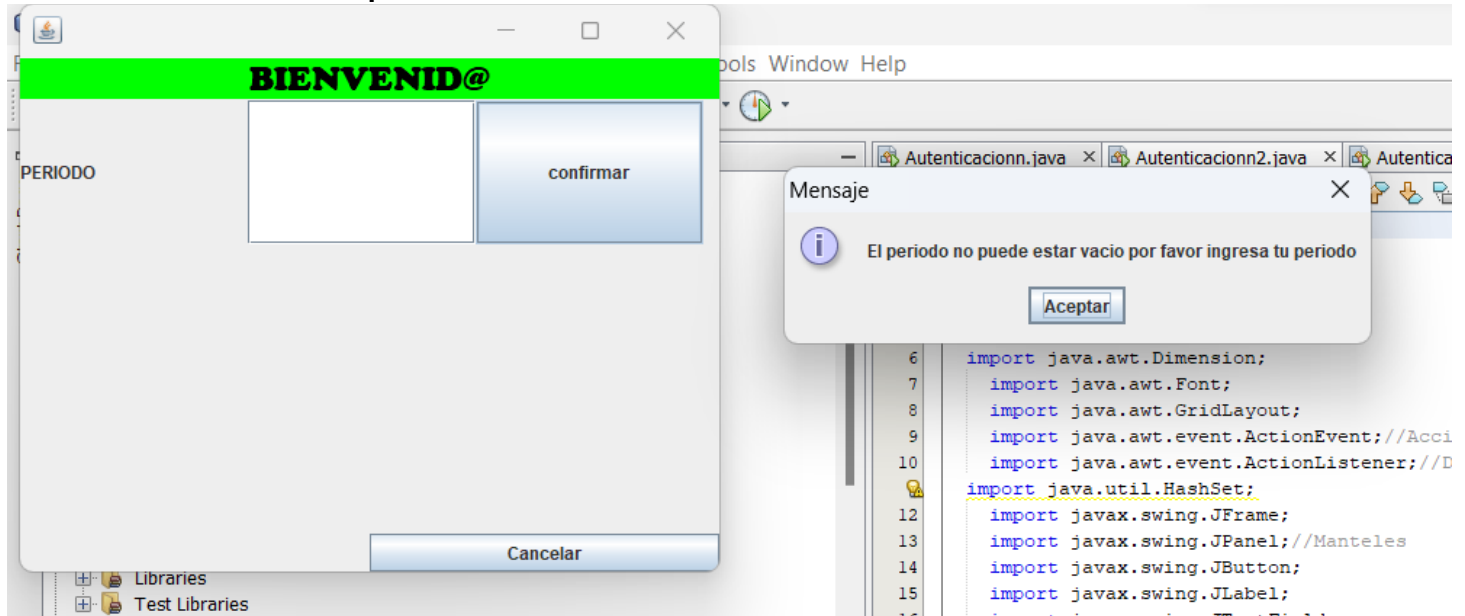



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x E
Source History
142     }
143     });
144
145
146     } // Termina el Constructor
147
148
149     // Metodo
150     public void borrar() {
151         periodo.setText("");
152     }
153 }
154 public void llamar() {
155
156     Autenticacionn3 terceraVentana= new Autenticacionn3();
157     terceraVentana.setVisible(true);
158     this.setVisible(false);
159 }
160
161
162
163
164
165
166 @Override
167 public void comprobar() {
168
169     try{
170         String periodo2= periodo.getText();
171         if(periodo2.isEmpty()){
172             throw new NullPointerException("El periodo ingresado está vacío");
173         }
174         if(periodo2.matches("\\d{4}-\\d{2}")){
175             confirmar.setVisible(false);
176             confirmar2.setVisible(true);
177             semestre.setVisible(true);
178             sem.setVisible(true);
179         }else{
180             JOptionPane.showMessageDialog(null, "Usted no ha ingresado un periodo valido");
181         }
182     } catch (NullPointerException d){
183         JOptionPane.showMessageDialog(null, "El periodo no puede estar vacío por favor ingresa tu periodo");
184     }
185 }
186
187
188
```

En esta sección utilicé la excepción (NullPointerException), la cual verificara que el usuario haya ingresado datos correctos, ya que si el campo donde se ingresara el periodo, si este esta vacío o nulo, el programa no puede hacer la validación del formato del periodo.



Resultado cuando la excepción entra en función:





Código de la ventana 3:

```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.
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,promedio;
27     private JPanel central,norte,sur;
28     int contador=0;
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);
47
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x
Source History
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 promedio=new JButton("Calcular Promedio");
64 promedio.addActionListener(manejador);
65
66
67 norte.add(titulo);
68 central.add(cal1);
69 central.add(calificacion1);
70
71 central.add(cal2);
72 central.add(calificacion2);
73
74 central.add(cal3);
75 central.add(calificacion3);
76 central.add(cal4);
77 central.add(calificacion4);
78 central.add(cal5);
79 central.add(calificacion5);
80 central.add(cal6);
81 central.add(calificacion6);
82 sur.add(aceptar);
83 sur.add(cancelar);
84 sur.add(promedio);
85
86
87 contenedor.add(norte, BorderLayout.NORTH);
88 contenedor.add(central, BorderLayout.CENTER);
89 contenedor.add(sur, BorderLayout.SOUTH);
90 titulo.setFont(new Font("Cooper Black",Font.PLAIN, 18));
91 titulo.setForeground(Color.black);
92 norte.setBackground(Color.green);
93 setSize(600,500); // pixeles, está es la mamá.
94 setVisible(true);
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x
Source History [Icons]
95 }
96 public static void main(String args[]) {
97     Autenticacionn3 ventana = new Autenticacionn3();
98     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
99 }
100 public void borrar() {
101     calificacion1.setText("");
102     calificacion2.setText("");
103     calificacion3.setText("");
104     calificacion4.setText("");
105     calificacion5.setText("");
106     calificacion6.setText("");
107 }
108
109 @Override
110 public void comprobar() {
111     try{
112         String c1 = calificacion1.getText();
113         int ca1 = Integer.parseInt(c1);
114         String c2 = calificacion2.getText();
115         int ca2 = Integer.parseInt(c2);
116         String c3 = calificacion3.getText();
117         int ca3 = Integer.parseInt(c3);
118         String c4 = calificacion4.getText();
119         int ca4 = Integer.parseInt(c4);
120         String c5 = calificacion5.getText();
121         int ca5 = Integer.parseInt(c5);
122         String c6 = calificacion6.getText();
123         int ca6 = Integer.parseInt(c6);
124
125         if((ca1 < 69 && ca1 > 0) ){
126             calificacion1.setText("N/A");
127
128         }if((ca2 < 69 && ca2 > 0) ){
129             calificacion2.setText("N/A");
130
131         }if((ca3 < 69 && ca3 > 0) ){
132             calificacion3.setText("N/A");
133
134         }if((ca4 < 69 && ca4 > 0) ){
135             calificacion4.setText("N/A");
136
137         }if((ca5 < 69 && ca5 > 0) ){
138             calificacion5.setText("N/A");
139
140         }if((ca6 < 69 && ca6 > 0) ){
141             calificacion6.setText("N/A");
142         }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java
Source History
142     }if((ca5<69 && ca5>0) ){
143         calificacion5.setText("N/A");
144     }if((ca6<69 && ca6>0) ){
145         calificacion6.setText("N/A");
146     }
147
148
149
150
151     }catch (NumberFormatException e){
152         JOptionPane.showMessageDialog(null,"Usted no ha ingresado valores numericos ");
153     }
154 }
155
156 @Override
157 public void visible(){
158     try{
159         String c1 = calificacion1.getText();
160         int cal = Integer.parseInt(c1);
161         String c2= calificacion2.getText();
162         int ca2= Integer.parseInt(c2);
163         String c3= calificacion3.getText();
164         int ca3= Integer.parseInt(c3);
165         String c4= calificacion4.getText();
166         int ca4= Integer.parseInt(c4);
167         String c5= calificacion5.getText();
168         int ca5= Integer.parseInt(c5);
169         String c6= calificacion6.getText();
170         int ca6= Integer.parseInt(c6);
171         if((cal>100) || (ca2>100) || (ca3>100) || (ca4>100) || (ca5>100) || (ca6>100)){
172             JOptionPane.showMessageDialog(null,"Usted ha ingresado valores mayores a 100 ");
173         }
174         if ((cal<0) || (ca2<0) || (ca3<0) || (ca4<0) || (ca5<0) || (ca6<0)) {
175             throw new IllegalArgumentException("El parámetro debe ser un valor positivo");
176         }
177     }
178 }
179
180
181     }catch(IllegalArgumentException b){
182         JOptionPane.showMessageDialog(null,"Error: Los valores ingresados no pueden ser menores a cero ");
183     }
184 }
185 public void calcular(){
186     try{
187         String c1 = calificacion1.getText();
188         int cal = Integer.parseInt(c1);
```

En esta sección utilice 3 excepciones(NumberFormatException,IllegalArgumentException y ArithmeticException), la primer excepción verificara que el usuario haya ingresado valores numéricos y no letras, para poder guardar calificaciones y calcular su promedio.

La segunda excepción se encarga de verificar que el usuario no ingrese valores negativos que afecten los cálculos del promedio, y la tercera excepción se encarga de verificar que el usuario no ingrese un 0 para calcular su promedio ya que la suma de todas sus calificaciones no puede ser dividida entre 0.



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x Exce
Source History
188 int ca1 = Integer.parseInt(c1);
189 String c2= calificacion2.getText();
190 int ca2= Integer.parseInt(c2);
191 String c3= calificacion3.getText();
192 int ca3= Integer.parseInt(c3);
193 String c4= calificacion4.getText();
194 int ca4= Integer.parseInt(c4);
195 String c5= calificacion5.getText();
196 int ca5= Integer.parseInt(c5);
197 String c6= calificacion6.getText();
198 int ca6= Integer.parseInt(c6);
199
200
201
202 int divisor;
203 divisor= Integer.parseInt(JOptionPane.showInputDialog("Introduce la cantidades de unidades a promediar"));
204 int promedioFinal=(ca1+ca2+ca3+ca4+ca5+ca6)/divisor;
205 JOptionPane.showMessageDialog(null,"Su promedio final es: "+promedioFinal);
206 }catch(ArithmeticException e){
207     JOptionPane.showMessageDialog(null,"Error: las calificaciones ingresadas no pueden ser divididas entre 0 ");
208 }
209
210
211
212
213 class Eventos implements ActionListener{
214
215     @Override
216     public void actionPerformed(ActionEvent ev) {
217         if(ev.getSource()==cancelar){
218             borrar();
219         }
220         if(ev.getSource()==aceptar){
221             contador++;
222             if(contador==1){
223                 visible();
224                 comprobar();
225             }
226
227             if(contador==2){
228
229                 JOptionPane.showMessageDialog(null,"Sus calificaciones se guardaron con exito ");
230                 System.exit(0);
231             }
232
233         }
234
235         if(ev.getSource()==promedio){
236             calcular();
237         }
238
239     }
240
241
242
243
244 }
245
246
247
248
249
250 }
251
```



Resultado de la primer excepción:

Ingrese las calificaciones de su materia seleccionada	
Ingrese su calificacion de la unidad 1	tre
Ingrese su calificacion de la unidad 2	89
Ingrese su calificacion de la unidad 3	tuub
Ingrese su calificacion de la unidad 4	fgt
Ingrese su calificacion de la unidad 5	89
Ingrese su calificacion de la unidad 6	89

Guardar calificaciones Cancelar Calcular Promedio

Mensaje
Usted no ha ingresado valores numericos
Aceptar

```
Autenticacionn.java x Autenticacionn2.java x A
Source History
217 if (ev.getSource () == cancel
==accepte
;
r==1) {
225 comprobar ();
226 }
227
228
229 if (contador==2) {
230
231 JOptionPane.s
232 System.exit (0)
233 }
234
235
236 }
```

Resultado de la segunda excepción:

Ingrese las calificaciones de su materia seleccionada	
Ingrese su calificacion de la unidad 1	-89
Ingrese su calificacion de la unidad 2	67
Ingrese su calificacion de la unidad 3	-76
Ingrese su calificacion de la unidad 4	90
Ingrese su calificacion de la unidad 5	56
Ingrese su calificacion de la unidad 6	45

Guardar calificaciones Cancelar Calcular Promedio

Mensaje
Error: Los valores ingresados no pueden ser menores a cero
Aceptar

```
Autenticacionn.java x Autenticacionn2.java x Autenticac
{
222 contador++;
223 if (contador==1) {
224 visible ();
225 comprobar ();
226 }
227
228
229 if (contador==2) {
230
231 JOptionPane.showMessa
232 System.exit (0);
233 }
234
235
236 }
```



Resultado de la tercer excepción:

The screenshot shows a Java application window titled "Ingrese las calificaciones de su materia seleccionada". The window contains a table for entering grades for six units. The grades entered are: Unidad 1: 56, Unidad 2: 78, Unidad 3: 99, Unidad 4: 67, Unidad 5: 88, and Unidad 6: 98. At the bottom of the window are three buttons: "Guardar calificaciones", "Cancelar", and "Calcular Promedio".

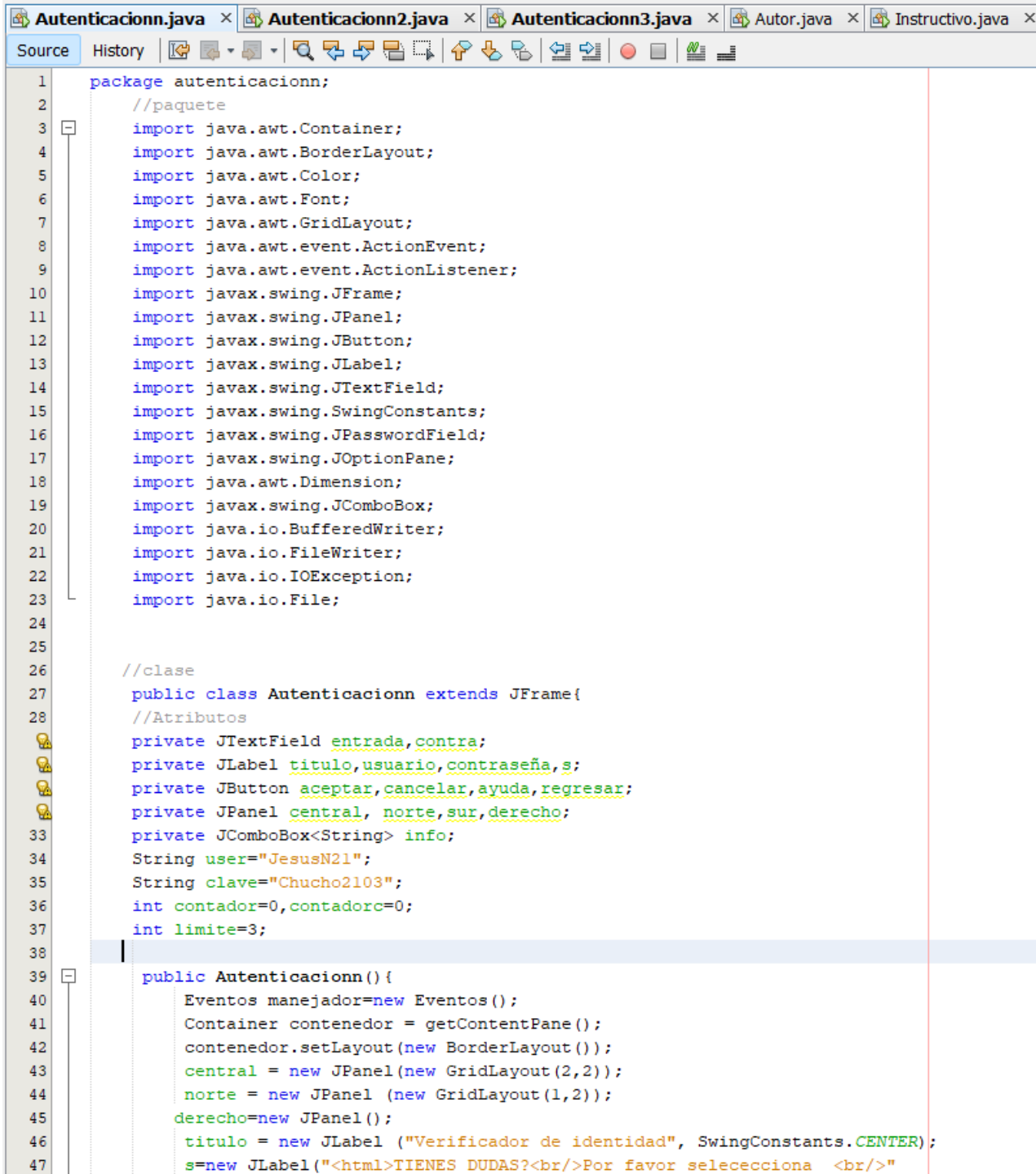
Ingrese las calificaciones de su materia seleccionada	
Ingrese su calificacion de la unidad 1	56
Ingrese su calificacion de la unidad 2	78
Ingrese su calificacion de la unidad 3	99
Ingrese su calificacion de la unidad 4	67
Ingrese su calificacion de la unidad 5	88
Ingrese su calificacion de la unidad 6	98

Buttons: Guardar calificaciones, Cancelar, Calcular Promedio

Overlaid on the application are two dialog boxes. The first is an "Entrada" (Input) dialog box with the text "Introduce la cantidades de unidades a promediar" and a text field containing "0". It has "Aceptar" and "Cancelar" buttons. The second is a "Mensaje" (Message) dialog box with an information icon and the text "Error: las calificaciones ingresadas no pueden ser divididas entre 0". It has an "Aceptar" button.

Unidad VI : Flujos y Archivos

Código de la ventana 1:



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
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 import java.io.BufferedWriter;
21 import java.io.FileWriter;
22 import java.io.IOException;
23 import java.io.File;
24
25
26 //clase
27 public class Autenticacionn extends JFrame{
28 //Atributos
29 private JTextField entrada,contra;
30 private JLabel titulo,usuario,contraseña,s;
31 private JButton aceptar,cancelar,ayuda,regresar;
32 private JPanel central,norte,sur,derecho;
33 private JComboBox<String> info;
34 String user="JesusN21";
35 String clave="Chucho2103";
36 int contador=0,contadorc=0;
37 int limite=3;
38
39 public Autenticacionn(){
40 Eventos manejador=new Eventos();
41 Container contenedor = getContentPane();
42 contenedor.setLayout(new BorderLayout());
43 central = new JPanel(new GridLayout(2,2));
44 norte = new JPanel (new GridLayout(1,2));
45 derecho=new JPanel();
46 titulo = new JLabel ("Verificador de identidad", SwingConstants.CENTER);
47 s=new JLabel("<html>TIENES DUDAS?<br/>Por favor seleccione <br/>")
```



```

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

48         + "la informacion <br/>que deseas conocer: <html>", SwingConstants.NORTH_EA
49         s.setVisible(false);
50         String[] elementos1 = {"Informacion del autor", "Instructivo"};
51         info= new JComboBox<>(elementos1);
52         info.setEditable(true);
53         ayuda= new JButton("?");
54         ayuda.setPreferredSize(new Dimension(45, 45));
55         ayuda.addActionListener(manejador);
56         regresar=new JButton("Volver al inicio");
57         regresar.setVisible(false);
58         regresar.addActionListener(manejador);
59         usuario= new JLabel ("Ingrese su usuario", SwingConstants.CENTER);
60         contraseña= new JLabel ("Ingrese su contraseña", SwingConstants.CENTER);
61         entrada=new JTextField(40);
62         contra= new JPasswordField(40);
63         sur= new JPanel(new GridLayout(1,1));
64         aceptar= new JButton("Aceptar");
65         aceptar.addActionListener(manejador);
66         cancelar= new JButton("Cancelar");
67         cancelar.addActionListener(manejador);
68         norte.add(titulo);
69         derecho.add(ayuda);
70         derecho.add(s);
71         derecho.add(info);
72         info.setVisible(false);
73         central.add(usuario);
74         central.add(entrada);
75         central.add(contraseña);
76         central.add(contra);
77         sur.add(aceptar);
78         sur.add(cancelar);
79         contenedor.add(norte, BorderLayout.NORTH);
80         contenedor.add(derecho, BorderLayout.EAST);
81         contenedor.add(central, BorderLayout.CENTER);
82         contenedor.add(sur, BorderLayout.SOUTH);
83         titulo.setFont(new Font("Cooper Black", Font.PLAIN, 18));
84         ayuda.setFont(new Font("Cooper Black", Font.PLAIN, 18));
85         titulo.setForeground(Color.black);
86         norte.setBackground(Color.green);
87         setSize(500,400); // pixeles, está es la mamá.
88         setVisible(true); // se define com un valor booleano.
89         info.addActionListener(e -> {
90             String opcionSeleccionada = (String) info.getSelectedItemAt();
91             if (opcionSeleccionada.equals("Informacion del autor")) {
92                 Autor cuartaVentana= new Autor();
93                 cuartaVentana.setVisible(true);
94                 this.setVisible(false);

```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
95         } else if(opcionSeleccionada.equals("Instructivo")) {
96             Instructivo quintaVentana= new Instructivo();
97             quintaVentana.setVisible(true);
98             this.setVisible(false);
99         }
100     });
101 } // termina el constructor.
102 public static void main(String args[]){
103     Autenticacionn ventana = new Autenticacionn();
104     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
105 }
106 public void borrar(){
107     entrada.setText("");
108     contra.setText("");
109 }
110 public void traer(){
111     Autenticacionn ven= new Autenticacionn();
112     ven.setVisible(true);
113     this.setVisible(false);
114 }
115 public class InvalidUserException extends Exception {
116     public InvalidUserException(String message) {
117         super(message);
118     }
119 }
120
121 public void verificarUsuario(){
122
123     try{
124
125         if(!user.equals(entrada.getText()) || !clave.equals(contra.getText())){
126             contador++;
127
128             entrada.setText("");
129             contra.setText("");
130
131             if(contador==3){
132                 System.exit(0);
133             }
134
135             throw new InvalidUserException("Usuario incorrecto");
136
137         }
138
139         Autenticacionn2 segundaVentana= new Autenticacionn2();
140         segundaVentana.setVisible(true);
141         this.setVisible(false);
142     }
143 }
```




```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x Excepciones.java x
Source History
String persona=JOptionPane.showInputDialog(null,"Antes de continuar por favor ingrese su nombre");
143 String rutaArchivo = "C:\\Users\\jesus\\Documents\\NetBeansProjects\\Autenticacionn\\src\\autenticacionn\\registro de calificaciones.txt";
144 try (BufferedWriter escritor = new BufferedWriter(new FileWriter(rutaArchivo, true))) {
145     File archivo = new File(rutaArchivo);
146     if (!archivo.exists()) {
147         archivo.createNewFile();
148     }
149     escritor.write("Nombre del alumno: "+persona);
150     escritor.newLine();
151 } catch (IOException e) {
152     System.out.println("Ocurrió un error al guardar los datos en el archivo.");
153     e.printStackTrace();
154 }
155
156 }catch(InvalidUserException e){
157     JOptionPane.showMessageDialog(null,"Contraseña o usuario incorrecto");
158 }
159
160 }
161
162
163 class Eventos implements ActionListener{
164     @Override
165     public void actionPerformed(ActionEvent ev) {
166         if(ev.getSource()==cancelar){
167             borrar();
168         }
169         if(ev.getSource()==aceptar){
170             verificarUsuario();
171         }
172     }
173     if(ev.getSource()==ayuda){
174         s.setVisible(true);
175         regresar.setVisible(true);
176         info.setVisible(true);
177         info.setPopupVisible(true);
178         contadorc++;
179         if(contadorc==2){
180             traer();
181         }
182     }
183 }
184 }
185 }
186
187 }
```

En este apartado agregue una variable, la cual me servirá para extraer el nombre del usuario e incluirlo en mi archivo, agregue una condición la cual creara el archivo automáticamente al recopilar los datos que queremos incluir en el archivo, importe algunas clases que me servirán para leer, escribir información en el archivo y verificar su existencia.



Resultado al correr el código con el usuario y contraseña correctos:

Verificador de identidad

Ingrese su usuario: JesusN21

Ingrese su contraseña:

Aceptar Cancelar

Input

Antes de continuar por favor ingrese su nombre

OK Cancel



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.io.BufferedWriter;
12 import java.io.File;
13 import java.io.FileWriter;
14 import java.io.IOException;
15 import java.util.HashSet;
16 import javax.swing.JFrame;
17 import javax.swing.JPanel; //Manteles
18 import javax.swing.JButton;
19 import javax.swing.JLabel;
20 import javax.swing.JTextField;
21 import javax.swing.JComboBox;
22 import javax.swing.SwingConstants;
23 import javax.swing.JOptionPane;
24 //Clase
25 public class Autenticacionn2 extends JFrame implements Validacion {
26 //Atributos
27 private JTextField periodo;
28 private JLabel texto, ciclo, semestre, asignatura;
29 private JButton aceptar, cancelar, confirmar, confirmar2;
30 private JPanel central, run;
31 private JPanel inicio;
32 private JPanel ultimo;
33 private JComboBox<String> materia, sem;
34
35
36 //Constructor
37
38 public Autenticacionn2 () {
39 Eventos manejador=new Eventos();
40 Container contenedor=getContentPane();
41 contenedor.setLayout(new BorderLayout());
42 central=new JPanel(new GridLayout(3,2));
43 inicio=new JPanel(new GridLayout(1,1));
44 ultimo=new JPanel(new GridLayout(1,1));
45 run=new JPanel();
46 inicio.setBackground(Color.green);
```



```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x
Source History
47 periodo =new JTextField(10);
48 ciclo=new JLabel("PERIODO");
49 semestre=new JLabel("SEMESTRE");
50 semestre.setVisible(false);
51 String[] elementos1 = {"Semestre 1", "Semestre 2", "Semestre 3","Semestre 4"
52     "Semestre 5","Semestre 6", "Semestre 7", "Semestre 8"};
53 sem= new JComboBox<>(elementos1);
54 sem.setVisible(false);
55
56 asignatura = new JLabel("MATERIA");
57 asignatura.setVisible(false);
58 String[] elementos2 = {"Fundamentos de programacion"};
59 materia= new JComboBox<>(elementos2);
60 materia.setVisible(false);
61 texto = new JLabel ("BIENVENID@", SwingConstants.CENTER);
62 aceptar=new JButton ("Aceptar");
63 aceptar.addActionListener(manejador);
64 aceptar.setVisible(false);
65 cancelar=new JButton ("Cancelar");
66 cancelar.addActionListener(manejador);
67 confirmar= new JButton("confirmar");
68 confirmar.addActionListener(manejador);
69 confirmar.setPreferredSize(new Dimension(45, 45));
70 confirmar2= new JButton("confirmar");
71 confirmar2.addActionListener(manejador);
72 confirmar2.setPreferredSize(new Dimension(45, 45));
73 confirmar2.setVisible(false);
74 inicio.add(texto);
75 run.add(confirmar);
76 central.add(ciclo);
77 central.add(periodo);
78 central.add(confirmar);
79 central.add(semestre);
80 central.add(sem);
81 central.add(confirmar2);
82 central.add(asignatura);
83 central.add(materia);
84 ultimo.add(aceptar);
85 ultimo.add(cancelar);
86 contenedor.add(run,BorderLayout.EAST);
87 contenedor.add(central,BorderLayout.CENTER);
88 contenedor.add(inicio,BorderLayout.NORTH);
89 contenedor.add(ultimo,BorderLayout.SOUTH);
90 setSize(500,400);
91 setVisible(true);
92 texto.setForeground(Color.BLACK);

```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java
Source History [Icons]
93 texto.setFont(new Font ("Cooper Black", Font.PLAIN,24));
94 sem.addActionListener(e -> {
95     String opcionSeleccionada = (String) sem.getSelectedItem();
96     if (opcionSeleccionada.equals("Semestre 1")) {
97
98         String[] subopciones1 = {"Fundamentos de programacion","Calculo Diferencial",
99             "Desarrollo sustentable","Matematicas Discretas","Quimica","Fundamentos de investigacion"};
100 materia.removeAllItems();
101 for (String subopcion : subopciones1) {
102     materia.addItem(subopcion);
103 }
104 } else if (opcionSeleccionada.equals("Semestre 2")) {
105     String[] subopciones2 = {"Programacion orientada a objetos","calculo integral",
106         "Taller de administracion","Algebra lineal","Probabilidad y estadistica","Fisica general"};
107 materia.removeAllItems();
108 for (String subopcion : subopciones2) {
109     materia.addItem(subopcion);
110 }
111 } else if (opcionSeleccionada.equals("Semestre 3")) {
112     String[] subopciones2 = {"Sistemas Operativos","Calculo vectorial","Estructura de datos",
113         "Telecomunicaciones","Investigacion de operaciones","Principios electronicos"};
114 materia.removeAllItems();
115 for (String subopcion : subopciones2) {
116     materia.addItem(subopcion);
117 }
118 } else if (opcionSeleccionada.equals("Semestre 4")) {
119     String[] subopciones2 = {"Arquitectura de computadoras","Ecuaciones diferenciales",
120         "Metodos Numericos","Temas avanzados","Sistemas operativos","Arquitectura de computadoras"};
121 materia.removeAllItems();
122 for (String subopcion : subopciones2) {
123     materia.addItem(subopcion);
124 }
125 } else if (opcionSeleccionada.equals("Semestre 5")) {
126     String[] subopciones2 = {"Redes de computadoras","Base de datos","Contabilidad financiera",
127         "Ingenieria de software","Lenguajes y Automatas","Simulacion"};
128 materia.removeAllItems();
129 for (String subopcion : subopciones2) {
130     materia.addItem(subopcion);
131 }
132 } else if (opcionSeleccionada.equals("Semestre 6")) {
133     String[] subopciones2 = {"Programacion web","Administracion de redes","Sistemas Programables",
134         "Ingenieria de software"};
135 materia.removeAllItems();
136 for (String subopcion : subopciones2) {
137     materia.addItem(subopcion);
138 }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x Excepciones.java x
Source History
139 } else if (opcionSeleccionada.equals("Semestre 7")) {
140     String[] subopciones2 = {"Internet de las cosas", "Proyectos de software", "Desarrollo de aplicaciones",
141         "Programacion logica", "Redes de datos"};
142     materia.removeAllItems();
143     for (String subopcion : subopciones2) {
144         materia.addItem(subopcion);
145     }
146 }
147 else {
148     String[] subopciones3 = {"Inteligencia artificial", "Ciberseguridad", "Programacion Reactiva",
149         "Taller de investigacion"};
150     materia.removeAllItems();
151     for (String subopcion : subopciones3) {
152         materia.addItem(subopcion);
153     }
154 }
155
156
157 String periodo= periodo.getText();
158
159 materia.addActionListener(o -> {
160     String mat= (String) materia.getSelectedItem();
161     String rutaArchivo = "C:\\Users\\jesus\\Documents\\NetBeansProjects\\Autenticacionn\\src\\autenticacionn\\registro de calificaciones.txt";
162     try (BufferedWriter escritor = new BufferedWriter(new FileWriter(rutaArchivo, true))) {
163         File archivo = new File(rutaArchivo);
164         if (!archivo.exists()) {
165             archivo.createNewFile();
166         }
167         escritor.write("Periodo: "+periodo+"\n"+"Semestre: "+opcionSeleccionada+"\n" + "Materia: " + mat );
168         escritor.newLine();
169     } catch (IOException l) {
170         System.out.println("Ocurrió un error al guardar los datos en el archivo.");
171         l.printStackTrace();
172     }
173 }
174 });
175 });
176
177 }
178 //Termina el Constructor
179
180
181 //Metodo
182 public void borrar(){
183     periodo.setText("");
184 }
```

En esta sección incluí un Action listener para poder extraer la información de las opciones seleccionadas por el usuario y guardarlas en una variable para después guardarlas dentro del archivo e incluí algunos saltos de línea entre la información recopilada para tener un poco más de organización en el archivo.



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x
Source History
185 }
186
187 public void llamar(){
188
189
190     Autenticacionn3 terceraVentana= new Autenticacionn3();
191     terceraVentana.setVisible(true);
192     this.setVisible(false);
193
194 }
195
196 @Override
197 public void comprobar(){
198
199     try{
200         String periodo2= periodo.getText();
201         if(periodo2.isEmpty()){
202             throw new NullPointerException("El periodo ingresado está vacío");
203         }
204         if(periodo2.matches("\\d{4}-\\d{2}")){
205
206             confirmar.setVisible(false);
207
208             confirmar2.setVisible(true);
209
210             semestre.setVisible(true);
211             sem.setVisible(true);
212         }else{
213             JOptionPane.showMessageDialog(null,"Usted no ha ingresado un periodo valido");
214         }
215     }catch(NullPointerException d){
216         JOptionPane.showMessageDialog(null,"El periodo no puede estar vacío por favor ingresa tu periodo");
217     }
218
219 }
220
221 @Override
222 public void visible(){
223     asignatura.setVisible(true);
224     materia.setVisible(true);
225     confirmar2.setVisible(false);
226     aceptar.setVisible(true);
227 }
228
229
230
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x
Source History [Icons]
231 public static void main(String args[]){
232     Autenticacionn2 ventana= new Autenticacionn2 ();
233     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
234 } //cierra método
235
236 class Eventos implements ActionListener{
237     @Override
238     public void actionPerformed(ActionEvent ev){
239
240         if(ev.getSource()==aceptar){
241
242             llamar();
243
244
245         }
246         if(ev.getSource()==cancelar){
247             borrar();
248         }
249         if(ev.getSource()==confirmar){
250
251
252             comprobar();
253
254         }
255         if(ev.getSource()==confirmar2){
256             visible();
257
258         }
259
260     }
261
262 }
263 }
```



Resultado al correr el código:



Código de la ventana 3:

```

Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.j
Source History
package autenticacionn;
import java.awt.Container;
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.GridLayout;
import java.awt.event.ActionEvent; //Acciones
import java.awt.event.ActionListener; //Detecta
import javax.swing.JFrame;
import javax.swing.JPanel; //Manteles
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JTextField;
import javax.swing.JComboBox;
import javax.swing.JOptionPane;
import javax.swing.SwingConstants;
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;
import java.io.File;

/**
 *
 * @author Jesus
 */
public class Autenticacionn3 extends JFrame implements Validacion {
    private JTextField calificacion1, calificacion2, calificacion3, calificacion4, calificacion5, calificacion6;
    private JLabel titulo, cal1, cal2, cal3, cal4, cal5, cal6;
    private JButton aceptar, cancelar, confirmar, confirmar2, promedio;
    private JPanel central, norte, sur;

    int contador=0;
    int promedioFinal;

    public Autenticacionn3() {
        Eventos manejador=new Eventos();
        Container contenedor = getContentPane();
        contenedor.setLayout(new BorderLayout());
        central = new JPanel(new GridLayout(6,2));
        norte = new JPanel (new GridLayout(1,1));
        titulo = new JLabel ("Ingrese las calificaciones de su materia seleccionada", SwingConstants.CENTER);
        cal1= new JLabel ("Ingrese su calificacion de la unidad 1",SwingConstants.CENTER);
        calificacion1=new JTextField(40);
        cal2= new JLabel ("Ingrese su calificacion de la unidad 2",SwingConstants.CENTER);
        calificacion2=new JTextField(40);
    }
}

```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Valida
Source History
47 cal3= new JLabel ("Ingrese su calificacion de la unidad 3",SwingConstants.CENTER);
48
49 calificacion3=new JTextField(40);
50
51 cal4= new JLabel ("Ingrese su calificacion de la unidad 4",SwingConstants.CENTER);
52
53 calificacion4=new JTextField(40);
54
55 cal5= new JLabel ("Ingrese su calificacion de la unidad 5",SwingConstants.CENTER);
56
57 calificacion5=new JTextField(40);
58
59 cal6= new JLabel ("Ingrese su calificacion de la unidad 6",SwingConstants.CENTER);
60
61 calificacion6=new JTextField(40);
62
63 sur= new JPanel(new GridLayout(1,1));
64 aceptar= new JButton("Guardar calificaciones");
65 aceptar.addActionListener(manejador);
66 cancelar= new JButton("Cancelar");
67 cancelar.addActionListener(manejador);
68 promedio=new JButton("Calcular Promedio");
69 promedio.addActionListener(manejador);
70
71
72 norte.add(titulo);
73 central.add(call);
74 central.add(calificacion1);
75
76 central.add(cal2);
77 central.add(calificacion2);
78
79 central.add(cal3);
80 central.add(calificacion3);
81 central.add(cal4);
82 central.add(calificacion4);
83 central.add(cal5);
84 central.add(calificacion5);
85 central.add(cal6);
86 central.add(calificacion6);
87 sur.add(aceptar);
88 sur.add(cancelar);
89 sur.add(promedio);
90
91
92 contenedor.add(norte, BorderLayout.NORTH);
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java
Source History
93 contenedor.add(central, BorderLayout.CENTER);
94 contenedor.add(sur, BorderLayout.SOUTH);
95 titulo.setFont(new Font("Cooper Black", Font.PLAIN, 18));
96 titulo.setForeground(Color.black);
97 norte.setBackground(Color.green);
98 setSize(600,500); // pixeles, está es la mamá.
99 setVisible(true);
100 }
101
102 public static void main(String args[]){
103     Autenticacionn3 ventana = new Autenticacionn3();
104     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
105
106 }
107 public void borrar(){
108     calificacion1.setText("");
109     calificacion2.setText("");
110     calificacion3.setText("");
111     calificacion4.setText("");
112     calificacion5.setText("");
113     calificacion6.setText("");
114 }
115
116 @Override
117 public void comprobar(){
118     try{
119         String c1 = calificacion1.getText();
120         int cal = Integer.parseInt(c1);
121         String c2= calificacion2.getText();
122         int ca2= Integer.parseInt(c2);
123         String c3= calificacion3.getText();
124         int ca3= Integer.parseInt(c3);
125         String c4= calificacion4.getText();
126         int ca4= Integer.parseInt(c4);
127         String c5= calificacion5.getText();
128         int ca5= Integer.parseInt(c5);
129         String c6= calificacion6.getText();
130         int ca6= Integer.parseInt(c6);
131
132
133         if((cal<=69 &&cal>0) ){
134             calificacion1.setText("N/A");
135
136
137         }if((ca2<=69 && ca2>0) ){
138             calificacion2.setText("N/A");
```




```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacio
Source History
139
140
141     }if((ca3<=69 && ca3>0) ){
142         calificacion3.setText("N/A");
143
144
145     }if((ca4<=69 && ca4>0) ){
146         calificacion4.setText("N/A");
147
148
149     }if((ca5<=69 && ca5>0) ){
150         calificacion5.setText("N/A");
151     }if((ca6<=69 && ca6>0) ){
152         calificacion6.setText("N/A");
153     }
154
155
156
157
158     }catch (NumberFormatException e){
159         JOptionPane.showMessageDialog(null,"Usted no ha ingresado valores numericos ");
160     }
161 }
162
163 @Override
164 public void visible(){
165     try{
166         String c1 = calificacion1.getText();
167         int cal = Integer.parseInt(c1);
168         String c2= calificacion2.getText();
169         int ca2= Integer.parseInt(c2);
170         String c3= calificacion3.getText();
171         int ca3= Integer.parseInt(c3);
172         String c4= calificacion4.getText();
173         int ca4= Integer.parseInt(c4);
174         String c5= calificacion5.getText();
175         int ca5= Integer.parseInt(c5);
176         String c6= calificacion6.getText();
177         int ca6= Integer.parseInt(c6);
178         if((cal>100) ||(ca2>100) ||(ca3>100) || (ca4>100) || (ca5>100) ||(ca6>100)){
179             JOptionPane.showMessageDialog(null,"Usted ha ingresado valores mayores a 100 ");
180         }
181         if ((cal<0) ||(ca2<0) ||(ca3<0) || (ca4<0) || (ca5<0) ||(ca6<0)) {
182             throw new IllegalArgumentException("El parámetro debe ser un valor positivo");
183         }
184     }
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x E
Source History
185 }
186
187
188 }catch(IllegalArgumentException b){
189     JOptionPane.showMessageDialog(null,"Error: Los valores ingresados no pueden ser menores a cero ");
190 }
191 }
192
193
194 public void calcular() {
195     try{
196         String c1 = calificacion1.getText();
197         int cal = Integer.parseInt(c1);
198         String c2= calificacion2.getText();
199         int ca2= Integer.parseInt(c2);
200         String c3= calificacion3.getText();
201         int ca3= Integer.parseInt(c3);
202         String c4= calificacion4.getText();
203         int ca4= Integer.parseInt(c4);
204         String c5= calificacion5.getText();
205         int ca5= Integer.parseInt(c5);
206         String c6= calificacion6.getText();
207         int ca6= Integer.parseInt(c6);
208
209
210
211         int divisor;
212         divisor= Integer.parseInt(JOptionPane.showInputDialog("Introduce la cantidades de unidades a promediar"));
213         promedioFinal=(cal+ca2+ca3+ca4+ca5+ca6)/divisor;
214
215         JOptionPane.showMessageDialog(null,"Su promedio final es: "+promedioFinal);
216     }catch(ArithmeticException e){
217         JOptionPane.showMessageDialog(null,"Error: las calificaciones ingresadas no pueden ser divididas entre 0 ");
218     }
219 }
220 public void calcular2() {
221
222     String c1 = calificacion1.getText();
223     int cal = Integer.parseInt(c1);
224     String c2= calificacion2.getText();
225     int ca2= Integer.parseInt(c2);
226     String c3= calificacion3.getText();
227     int ca3= Integer.parseInt(c3);
228     String c4= calificacion4.getText();
229     int ca4= Integer.parseInt(c4);
230     String c5= calificacion5.getText();
```



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x Excepciones.java x
Source History
231     int ca5= Integer.parseInt(c5);
232     String c6= calificacion6.getText();
233     int ca6= Integer.parseInt(c6);
234
235     promedioFinal=(ca1+ca2+ca3+ca4+ca5+ca6)/6;
236
237 }
238 public void datos(){
239     String registro1= calificacion1.getText();
240     String registro2= calificacion2.getText();
241     String registro3= calificacion3.getText();
242     String registro4= calificacion4.getText();
243     String registro5= calificacion5.getText();
244     String registro6= calificacion6.getText();
245
246     String rutaArchivo = "C:\\Users\\jesus\\Documents\\NetBeansProjects\\Autenticacionn\\src\\autenticacionn\\registro de calificaciones.txt";
247     try (BufferedWriter escritor = new BufferedWriter(new FileWriter(rutaArchivo, true))) {
248         File archivo = new File(rutaArchivo);
249         if (!archivo.exists()) {
250             archivo.createNewFile();
251         }
252         escritor.write("Calificacion de la unidad 1: "+"\\t"+registro1 + "\\n" + "Calificacion de la unidad 2: "+"\\t" + registro2 + "\\n" +
253             "Calificacion de la unidad 3: "+"\\t" + registro3+ "\\n" + "Calificacion de la unidad 4: "+"\\t" + registro4 + "\\n" +
254             "Calificacion de la unidad 5: "+"\\t"+registro5+"\\n" + "Calificacion de la unidad 6: "+"\\t"+registro6+
255             "\\n"+"Su promedio final es: "+promedioFinal+"\\n" + "\\n" );
256     } catch (IOException e) {
257         System.out.println("Ocurrió un error al guardar los datos en el archivo.");
258         e.printStackTrace();
259     }
260
261 }
262
263 class Eventos implements ActionListener{
264
265     @Override
266     public void actionPerformed(ActionEvent ev) {
267         if(ev.getSource()==cancelar){
268             borrar();
269         }
270         if(ev.getSource()==aceptar){
271
272             contador++;
273             if(contador==1){
274                 visible();
275                 calcular2();
276                 comprobar();
277             }
278         }
279     }
280 }
```

En esta parte del código extraje la información de las variables donde se almacenan las calificaciones y el promedio para añadirlas al archivo, utilice algunas herramientas como tabuladores y saltos de línea para organizar la información, así como un escritor.write el cual me permitirá añadir datos al archivo según las variables que requiera.



```
277
278
279         }
280
281         if(contador==2) {
282
283             JOptionPane.showMessageDialog(null,"Sus calificaciones se guardaron con éxito ");
284             datos();
285
286             System.exit(0);
287
288         }
289
290     }
291     if(ev.getSource()==promedio) {
292         calcular();
293     }
294
295 }
296
297 }
298
299 }
300
```

Resultado al correr el código:

Ingrese las calificaciones de su materia seleccionada

Ingrese su calificacion de la unidad 1

Ingrese su calificacion de la unidad 2

Ingrese su calificacion de la unidad 3

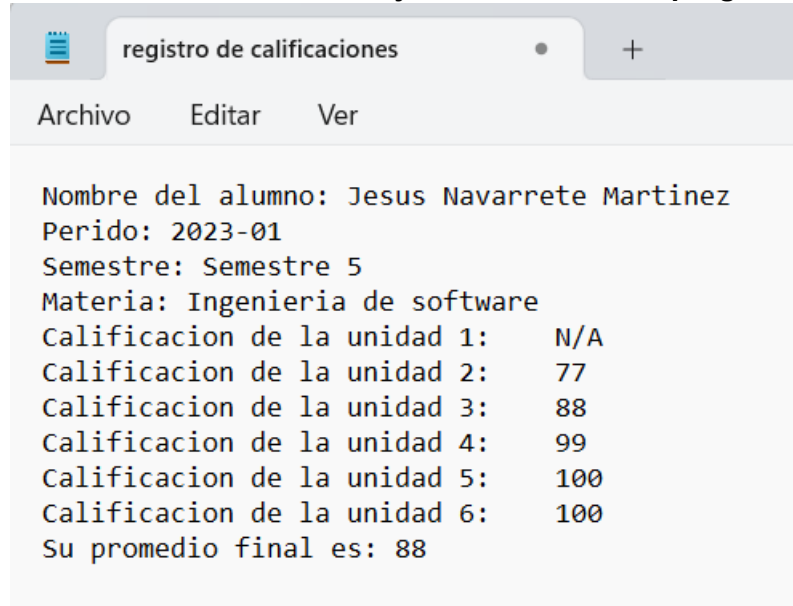
Ingrese su calificacion de la unidad 4

Ingrese su calificacion de la unidad 5

Ingrese su calificacion de la unidad 6

Guardar calificaciones Cancelar Calcular Promedio

Archivo creado durante la ejecución de todo el programa:



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.

Durante el desarrollo e implementación de los distintos tipos de excepciones en java me pude dar cuenta que las excepciones son de vital importancia para detectar y manejar errores que pueden llegar a ocurrir en nuestro programa durante su ejecución y su interacción con el usuario, ya que durante el desarrollo del programa podemos llegar a identificar aquellos bloques de código que son más susceptibles a los errores de ejecución, por lo tanto podemos capturar los errores que pueden surgir y tomar las medidas adecuadas, como indicarle al usuario que los valores ingresados no son correctos o que hay algunos elementos a los que no les ha proporcionado algún valor, esto ayudara al usuario a darse cuenta de sus errores para corregirlos y así el programa pueda realizar sus funciones sin problema alguno lo que mejora considerablemente la eficiencia del programa que hemos desarrollado.

Con la implementación de archivos al programa que estuvimos desarrollando podemos darnos cuenta que los flujos y archivos se consideran herramientas esenciales para guardar, leer, escribir y transferir datos de manera más sencilla y confiable además también nos permiten organizar la información que vamos ingresando según las necesidades que tengamos, además lo podemos lograr de una manera muy eficiente, sin embargo hay que ser muy cuidadosos con las rutas que seleccionamos para guardar la información que el usuario ira ingresando en nuestro programa ya que tratar de acceder al archivo para escribir o consultar información en una ruta donde el archivo no existe o no ha sido generado puede ocasionar múltiples errores como perdida de información ya que no tendría donde guardar los datos.

Los flujos y archivos nos permiten interactuar con el usuario y la información que ingresara en el programa, pues además nos permiten tener un control más detallado de los datos recopilados, y así mismo poder consultarlos o acceder a ellos en algún otro momento cuando la situación así lo requiera, con ello podemos visualizar la importancia de una buena organización y el almacenamiento seguro de la información.