

Nombre de la práctica	UNIDAD 5.Excepciones			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         if(ev.getSource()==limpiar){
128             borrar();
129         }
130         if(ev.getSource()==salir){
131             System.exit(0);
132         }
133     }
134 }
135
136
137
138
139
140
```



Resultado de correr el código Encriptador :



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 });

```



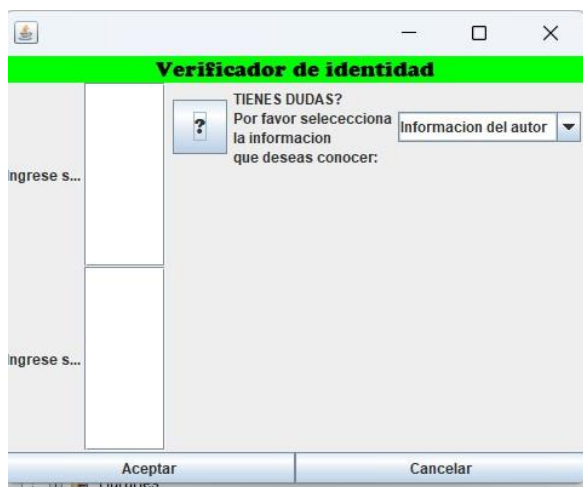
```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.ja
Source History
95 // termina el constructor.
96 public static void main(String args[]){
97     Autenticacionn ventana = new Autenticacionn();
98     ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
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     }catch(InvalidUserException e){
138
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 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     }
}
```

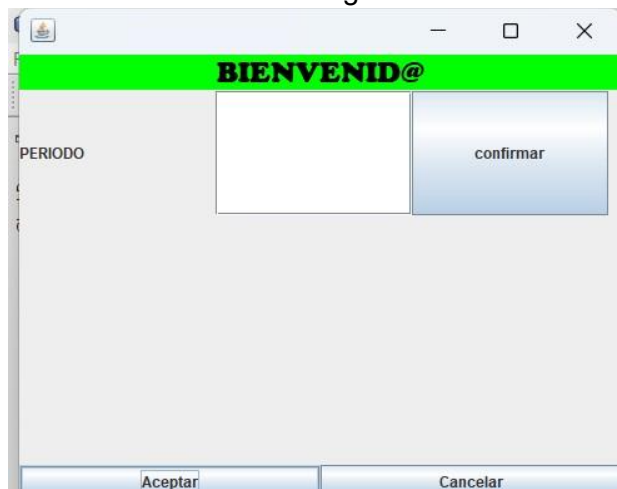



```
Autenticacionn.java x Autenticacionn2.java x Autenticacionn3.java x Autor.java x Instructivo.java x Validacion.java x Abs3203.java x E
Source History
142 }
143 });
144
145
146 //Termina el Constructor
147
148
149 //Metodo
150 public void borrar() {
151     periodo.setText("");
152 }
153
154 public 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 válido");
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 | [Icons]
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
236         if(ev.getSource()==promedio){
237             calcular();
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 está en un fondo verde. El cuerpo de la ventana tiene un fondo gris y contiene seis filas, cada una con un texto a la izquierda y un cuadro de entrada a la derecha. Los textos son: "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" y "Ingrese su calificacion de la unidad 6". En la parte inferior de la ventana hay tres botones: "Guardar calificaciones", "Cancelar" y "Calcular Promedio".

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



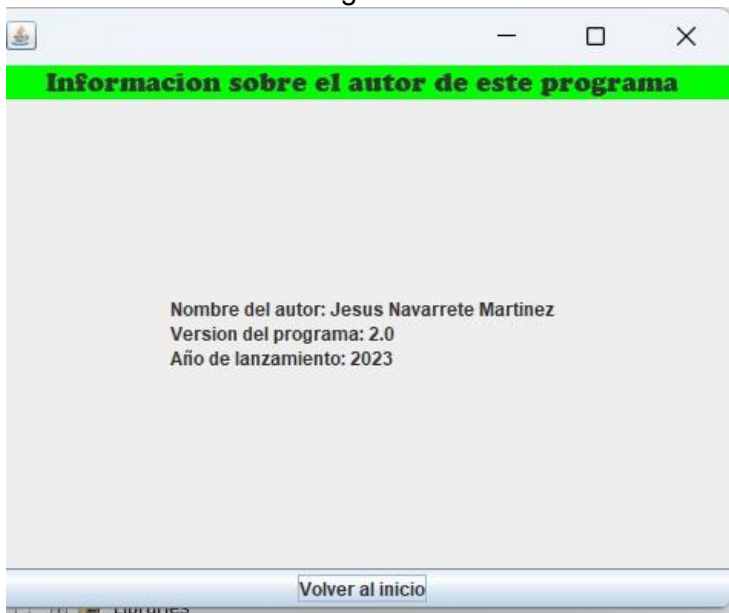
Código de la ventana 4:

```
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/>"

```



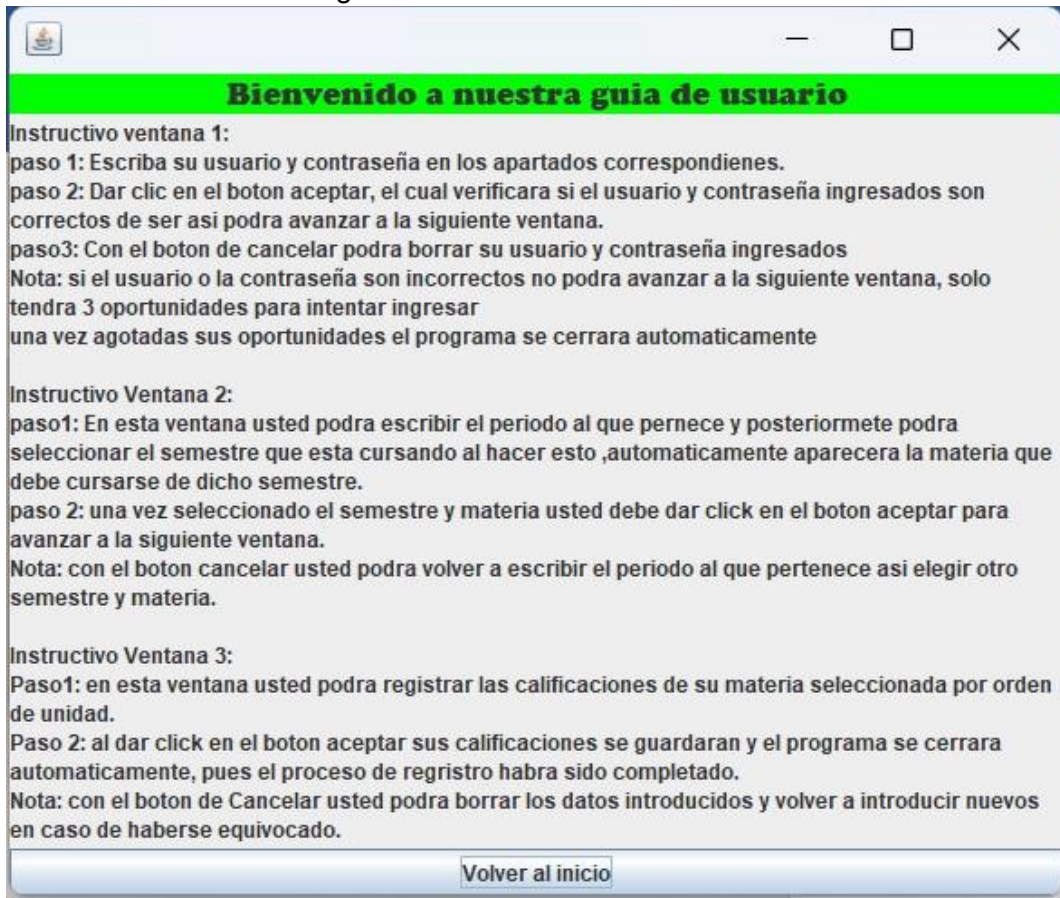
```

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 }
```



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.