



**INSTITUTO TECNOLÓGICO NACIONAL DE MÉXICO**  
**INSTITUTO TECNOLÓGICO DE TLAXIACO**

**DEPT. CIENCIAS BÁSICAS**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**TÓPICOS AVANZADOS DE PROGRAMACIÓN (4US)**

**PROYECTO FINAL**

**20620258 MARÍA GUADALUPE FELIPE HERNÁNDEZ**

**ING. ROMAN CRUZ JOSE ALFREDO**

**6 SEMESTRE**

**martes 29 de mayo de 2023**



## CONTENIDO

INTRODUCCIÓN.....	5
OBJETIVO DEL SISTEMA.....	5
OBJETIVOS ESPECÍFICOS .....	5
JUSTIFICACIÓN .....	5
PLANTEAMIENTO DEL PROBLEMA.....	6
PROTOTIPO .....	7
DESARROLLO.....	8
CODIFICACIÓN.....	12
PRUEBAS MANUALES .....	25
COLUSIÓN.....	28



## Tabla de ilustraciones

Ilustración 1 Figma Inicio y Registro.....	7
Ilustración 2 Figma Menú .....	8
Ilustración 3 Figma Almacén .....	8
Ilustración 4 Crear form1 Inicio.....	9
Ilustración 5 Crear siguiente Form2 Registro .....	9
Ilustración 6 Crear Form3 Categoría .....	10
Ilustración 7 Crear Form4 Inventario venta .....	10
Ilustración 8 Crear base de datos en MySQL.....	11
Ilustración 9 Crear tablas .....	11
Ilustración 10 Ingresar datos de las tablas .....	11
Ilustración 11 Clase conexión.....	12
Ilustración 12 Clase conexión MySQL.....	12
Ilustración 13 Clase Usuario.....	13
Ilustración 14 Clase Producto.....	13
Ilustración 15 Clase ProductoConsulta .....	14
Ilustración 16 Clase ProductoConsulta 2 .....	14
Ilustración 17 Clase ProductoConsulta 3 .....	15
Ilustración 18 Clase Commun (Imagen) .....	15
Ilustración 19 Código de form1 (Inicio).....	16
Ilustración 20 Código de form1 (Inicio) 2.....	16
Ilustración 21 Código de form1 (Inicio) 3.....	17
Ilustración 22 Código de Form2 Registro .....	17
Ilustración 23 Código de Form3 Categoría.....	18
Ilustración 24 Código de Form4 Inventario venta .....	19
Ilustración 25 Código de Form4 Inventario venta 2 .....	19
Ilustración 26 Código de Form4 Inventario venta 3 .....	20
Ilustración 27 Código de Form4 Inventario venta 4 .....	21
Ilustración 28 Código de Form4 Inventario venta 5 .....	21
Ilustración 29 Código de Form4 Inventario venta 6 .....	22
Ilustración 30 Código de Form4 Inventario venta 7 .....	23
Ilustración 31 Código de Form4 Inventario venta 8 .....	23
Ilustración 32 Código de Form4 Inventario venta 9 .....	24
Ilustración 33 Código de Form4 Inventario venta 10 .....	24
Ilustración 34 inicio.....	25
Ilustración 35 mensaje de error inicio .....	25
Ilustración 36 registro .....	25
Ilustración 37 Ingresar Inicio.....	25
Ilustración 38 Registro gurdado.....	25
Ilustración 39 Mensaje de bienvenida .....	25
Ilustración 40 Menú .....	26
Ilustración 41 Alta de Producto.....	26
Ilustración 42 Producto guardado.....	27



Ilustración 43 Producto registrado en BD .....	27
Ilustración 44 Venta de Producto .....	27
Ilustración 45 Pago Efectivo .....	28
Ilustración 46 Mensaje de éxito de Cobro .....	28





## INTRODUCCIÓN

En el presente informe se describe la práctica realizada en el desarrollo e implementación de un sistema de ventas para. El objetivo de esta práctica fue diseñar y poner en funcionamiento un sistema que agilizará y optimizará los procesos de venta de la empresa, brindando una mejor experiencia tanto para los clientes como para el personal.

## OBJETIVO DEL SISTEMA

El objetivo principal del sistema de ventas implementado fue mejorar la eficiencia y la precisión en el proceso de ventas de la empresa. Además de automatizar tareas y reducir errores, el sistema tenía como finalidad guardar datos de usuarios nuevos, gestionar un almacenamiento para ingresar, modificar, buscar y eliminar productos, así como un cajero de compras.

## OBJETIVOS ESPECÍFICOS

1. Agilizar el proceso de registro de usuarios: El objetivo es implementar un sistema que permita un rápido y sencillo registro de nuevos usuarios, capturando la información relevante y generando perfiles de usuario de manera eficiente.
2. Optimizar la gestión de inventario: El objetivo es desarrollar un sistema que permita llevar un control preciso de los productos en existencia.
3. Facilitar la gestión de caja sin factura: El objetivo es implementar un sistema que permita proporcionar una interfaz intuitiva para el registro de ventas, la gestión del efectivo recibido y el cambio proporcionado al cliente.

## JUSTIFICACIÓN



Boulevard Tecnológico Km. 2.5, Llano Yosovee C.P. 69800. Tlaxiaco, Oaxaca. Tel. (953) 55 21322 y (953) 55 20405, e-mail: [dir\\_tlaxiaco@tecnm.mx](mailto:dir_tlaxiaco@tecnm.mx) [tecnm.mx](http://tecnm.mx) | [tlaxiaco.tecnm.mx](http://tlaxiaco.tecnm.mx)



**2022** *Ricardo Flores*  
Año de *Magón*  
PRECURSOR DE LA REVOLUCIÓN MEXICANA



La implementación de un sistema de registro de usuario, inventario y caja se justifica por varias razones fundamentales:

**Eficiencia operativa:** Un sistema automatizado y centralizado para el registro de usuarios, la gestión del inventario y la caja permitirá agilizar y simplificar los procesos internos de la empresa. Esto reducirá la carga de trabajo manual, minimizará los errores humanos y mejorará la eficiencia en general.

**Control y organización del inventario:** Un sistema de gestión de inventario permitirá llevar un control más preciso y detallado de los productos disponibles. Además, un mejor control del inventario permitirá una planificación más precisa y una reducción de costos innecesarios.

**Transparencia y precisión en la gestión de la caja:** La implementación de un sistema de caja permitirá un registro claro y detallado de las transacciones en efectivo sin generar una factura formal. Además, un sistema automatizado de caja precisos sobre los ingresos diarios, facilitando la toma de decisiones financieras y la identificación de tendencias y oportunidades de mejora.

**Mejor experiencia del cliente:** Un sistema eficiente de registro de usuarios y gestión de inventario garantizará una atención rápida y precisa a los clientes, reduciendo los tiempos de espera y evitando errores en los datos registrados. Además, la implementación de un sistema de caja agilizar el proceso de pago, evitando la necesidad de generar una factura formal en ciertos casos.

La implementación de un sistema de registro de usuario, inventario y caja sin factura traerá consigo beneficios significativos, tales como una mayor eficiencia operativa, un control más preciso del inventario, una gestión transparente de la caja y una mejora en la experiencia del cliente. Estos beneficios contribuirán a fortalecer la competitividad y el éxito a largo plazo de la empresa.

## PLANTEAMIENTO DEL PROBLEMA

Antes de la implementación del sistema de registro de usuario, inventario y caja sin factura, la empresa se enfrentaba a diversos desafíos en sus operaciones diarias. Entre ellos se encuentran:

Ineficiencia en el proceso de registro de nuevos usuarios, lo cual resultaba en largos tiempos de espera y pérdida de información importante.

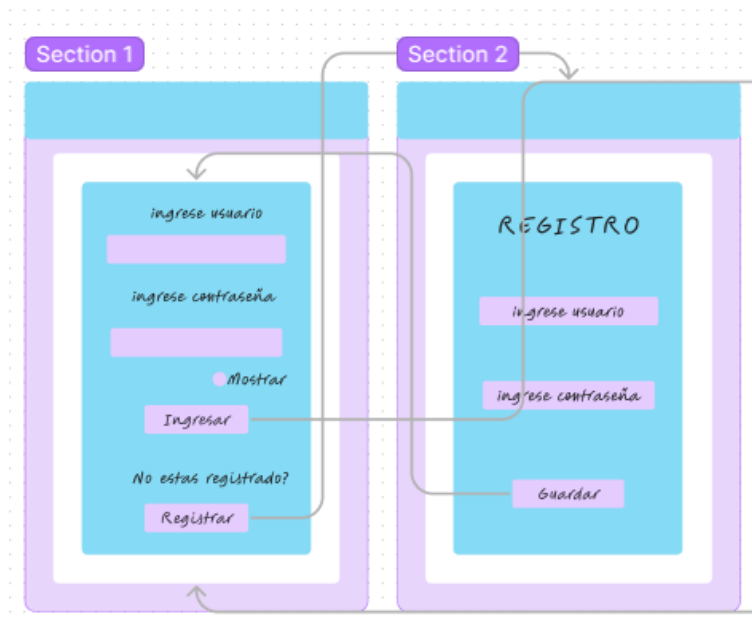


Dificultades para mantener un control preciso del inventario, lo que llevaba a errores en la gestión de existencias, pérdida de ventas por falta de productos.

Carencia de un sistema adecuado para gestionar las transacciones en efectivo, lo que generaba confusión en la caja y dificultaba la conciliación de los ingresos.

Estos problemas impactaban negativamente en la eficiencia operativa, la satisfacción del cliente y la capacidad de tomar decisiones informadas. Por lo tanto, se plantea la necesidad de implementar un sistema que resuelva estos desafíos y mejore la gestión de usuarios, inventario y ventas en caja.

## PROTOTIPO



*Ilustración 1 Figma Inicio y Registro*

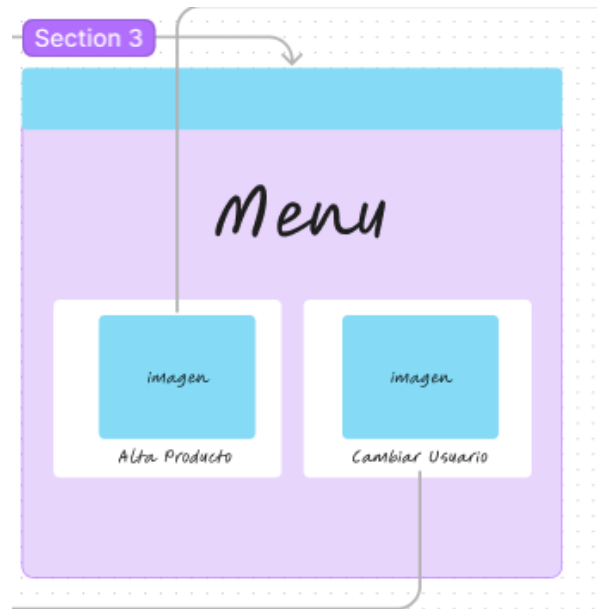


Ilustración 2 Figma Menú

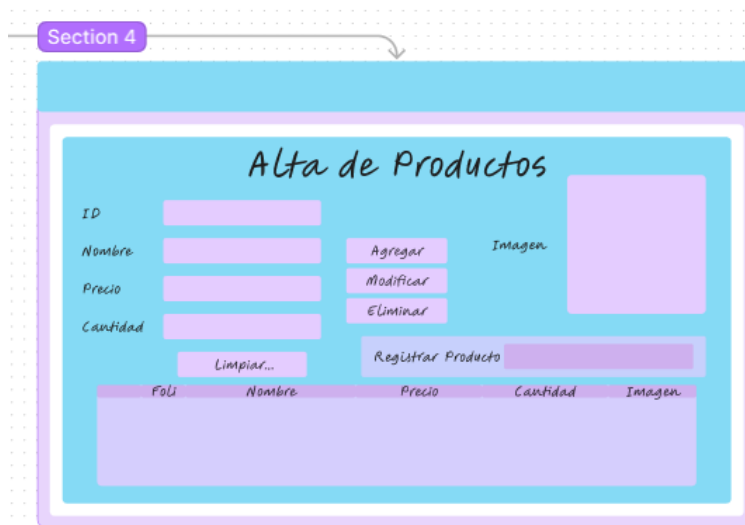


Ilustración 3 Figma Almacén

## DESARROLLO



Boulevard Tecnológico Km. 2.5, Llano Yosovee C.P. 69800. Tlaxiaco, Oaxaca. Tel. (953) 55 21322 y (953) 55 20405, e-mail: [dir\\_tlaxiaco@tecnm.mx](mailto:dir_tlaxiaco@tecnm.mx) [tecnm.mx](http://tecnm.mx) | [tlaxiaco.tecnm.mx](http://tlaxiaco.tecnm.mx)



**2022** *Ricardo Flores Magón*  
Año de Magón  
PRECURSOR DE LA REVOLUCIÓN MEXICANA





Formulario de inicio de sesión con los siguientes campos y botones:

- Campo de texto: *Ingrese Usuario*
- Campo de texto: *Ingrese Contraseña*
- Botón: *Ingresar*
- Enlace: *¿No está Registrado? Registrarse*

Ilustración 4 Crear form1 Inicio



Formulario de registro con los siguientes campos y botones:

- Campo de texto: *Ingrese Usuario*
- Campo de texto: *Ingrese Contraseña*
- Botón: *Guardar*

Ilustración 5 Crear siguiente Form2 Registro

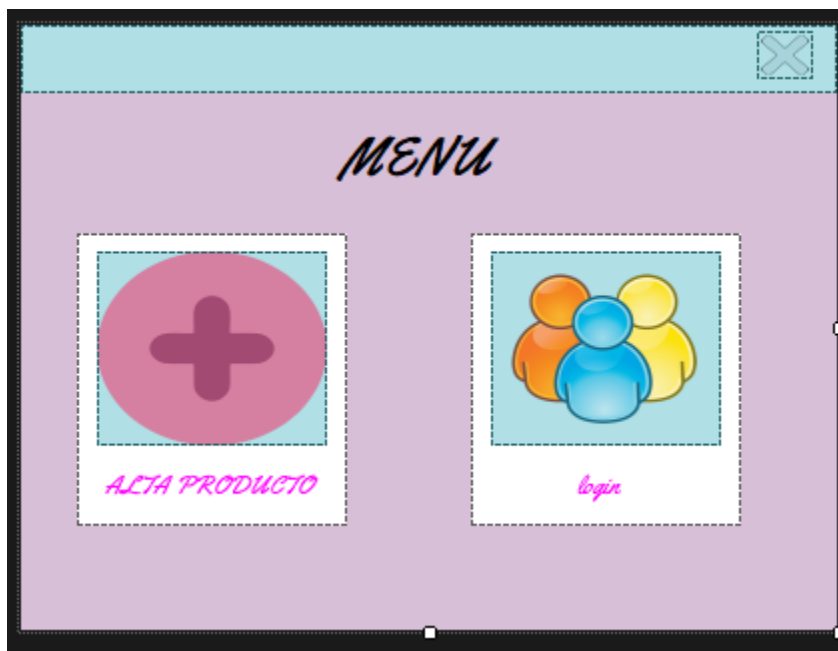


Ilustración 6 Crear Form3 Categoría

### ALTA DE PRODUCTOS

ID:

NOMBRE:

PRECIO:

CANTIDAD:

Folio	Nombre	Precio	Cantidad	Imagen

IMAGEN

### VENTA PRODUCTOS

Cantidad Producto:

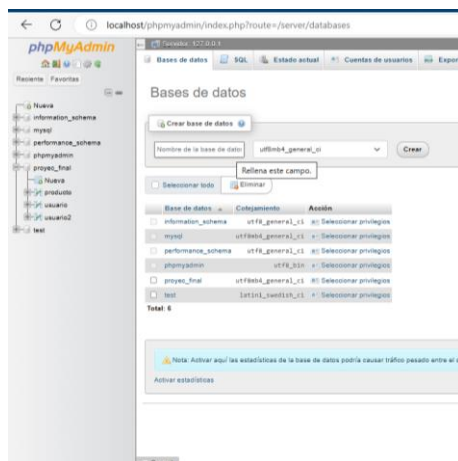
Folio	Nombre	Precio	CantidadP	Sub Total

Total Pagar \$:

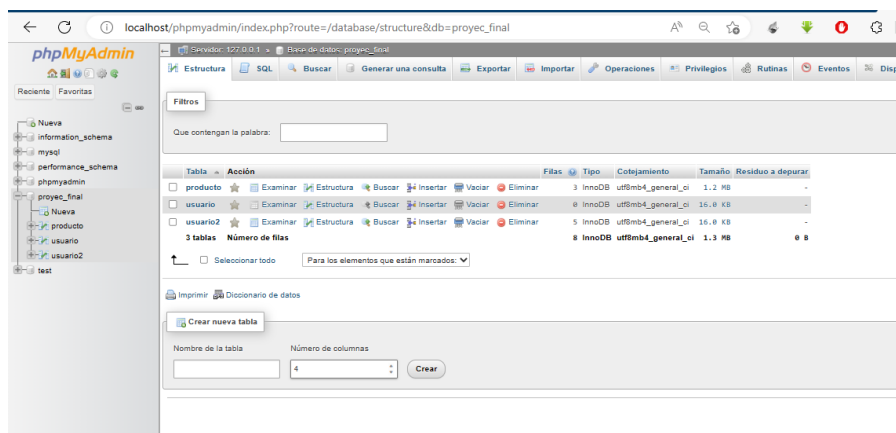
Efectivo:

Cambio:

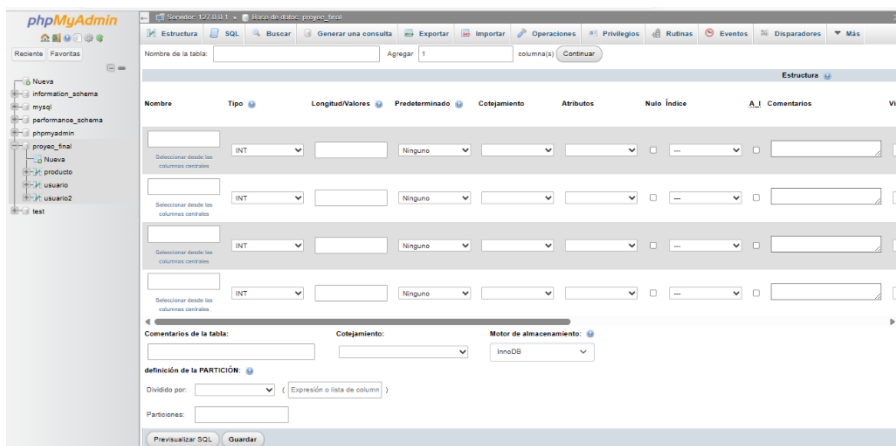
Ilustración 7 Crear Form4 Inventario venta



*Ilustración 8 Crear base de datos en MySQL*



*Ilustración 9 Crear tablas*



*Ilustración 10 Ingresar datos de las tablas*

## CODIFICACIÓN

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace projfinal
8 {
9     1 referencia
10     internal class Conexion
11     {
12         protected string server = "localhost";
13         protected string database = "proyec_final";
14         protected string user = "root";
15         protected string password = "";
16     }
17 }
```

*Ilustración 11 Clase conexión*

```
8
9 namespace projfinal
10 {
11     3 referencias
12     internal class ConexionMySQL : Conexion
13     {
14         private MySqlConnection conexion;
15         private string cadenaConexion;
16
17         1 referencia
18         public ConexionMySQL()
19         {
20             cadenaConexion = "Database=" + database +
21             "; DataSource=" + server +
22             "; User Id=" + user +
23             "; Password=" + password;
24             conexion = new MySqlConnection(cadenaConexion);
25         }
26
27         4 referencias
28         public MySqlConnection getConexion()
29         {
30             try
31             {
32                 if (conexion.State != System.Data.ConnectionState.Open)
33                 {
34                     conexion.Open();
35                 }
36             }
37             catch (Exception e)
38             {
39                 MessageBox.Show(e.ToString());
40             }
41             return conexion;
42         }
43
44         1 referencia
45         public void closeConexion()
46         {
47             conexion.Close();
48         }
49     }
50 }
```

*Ilustración 12 Clase conexión MySQL*

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace projfinal
8 {
9     12 referencias
10     internal class Producto
11     {
12         5 referencias
13         public int producto_id { get; set; }
14         5 referencias
15         public string nombre { get; set; }
16         5 referencias
17         public float precio { get; set; }
18         5 referencias
19         public int cantidad { get; set; }
20         5 referencias
21         public byte[] imagen { get; set; }
22     }
23 }
```

*Ilustración 13 Clase Usuario*

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace projfinal
8 {
9     12 referencias
10     internal class Producto
11     {
12         5 referencias
13         public int producto_id { get; set; }
14         5 referencias
15         public string nombre { get; set; }
16         5 referencias
17         public float precio { get; set; }
18         5 referencias
19         public int cantidad { get; set; }
20         5 referencias
21         public byte[] imagen { get; set; }
22     }
23 }
```

*Ilustración 14 Clase Producto*

```

1  using MySql.Data.MySqlClient;
2  using System;
3  using System.Collections.Generic;
4  using System.Linq;
5  using System.Text;
6  using System.Threading.Tasks;
7  using System.Windows.Forms;
8
9  namespace projfinal
10 {
11     3 referencias
12     internal class ProductoConsultas
13     {
14         private ConexionMySQL conexionMySQL;
15         private List<Producto> mProductos;
16
17         1 referencia
18         public ProductoConsultas()
19         {
20             conexionMySQL = new ConexionMySQL();
21             mProductos = new List<Producto>();
22         }
23
24         1 referencia
25         public List<Producto> getProductos(string filtro)
26         {
27             string QUERY = "SELECT * FROM producto";
28
29             MySqlCommand mCommand = new MySqlCommand(QUERY);
30             mCommand.Connection = conexionMySQL.getConexion();
31             mReader = mCommand.ExecuteReader();
32
33             while (mReader.Read())
34             {
35                 mProducto = new Producto();
36                 mProducto.producto_id = mReader.GetInt16("producto_id");
37                 mProducto.nombre = mReader.GetString("nombre");
38                 mProducto.precio = mReader.GetInt16("precio");
39                 mProducto.cantidad = mReader.GetInt16("cantidad");
40                 mProducto.imagen = (byte[])mReader.GetValue(4);
41                 mProductos.Add(mProducto);
42             }
43             mReader.Close();
44
45             catch (MySqlException ex)
46             {
47                 MessageBox.Show(ex.Message);
48             }
49             finally
50             {
51                 conexionMySQL.closeConexion();
52             }
53
54             return mProductos;
55         }
56
57         1 referencia
58         public bool agregarProducto(Producto mProducto)
59         {
60             string INSERT = "INSERT INTO producto (nombre, precio, cantidad, imagen)" + " values (@nombre, @precio, @cantidad, @imagen)";
61
62             MySqlCommand mCommand = new MySqlCommand(INSERT);
63             mCommand.Connection = conexionMySQL.getConexion();
64             mCommand.Parameters.AddWithValue("@nombre", mProducto.nombre);
65             mCommand.Parameters.AddWithValue("@precio", mProducto.precio);
66             mCommand.Parameters.AddWithValue("@cantidad", mProducto.cantidad);
67             mCommand.Parameters.AddWithValue("@imagen", mProducto.imagen);
68             mCommand.ExecuteNonQuery();
69         }
70     }
71 }

```

Ilustración 15 Clase ProductoConsulta

```

36         "precio LIKE '%" + filtro + "%' OR " +
37         "cantidad LIKE '%" + filtro + "%'";
38     }
39
40     MySqlCommand mCommand = new MySqlCommand(QUERY);
41     mCommand.Connection = conexionMySQL.getConexion();
42     mReader = mCommand.ExecuteReader();
43
44     while (mReader.Read())
45     {
46         mProducto = new Producto();
47         mProducto.producto_id = mReader.GetInt16("producto_id");
48         mProducto.nombre = mReader.GetString("nombre");
49         mProducto.precio = mReader.GetInt16("precio");
50         mProducto.cantidad = mReader.GetInt16("cantidad");
51         mProducto.imagen = (byte[])mReader.GetValue(4);
52         mProductos.Add(mProducto);
53     }
54     mReader.Close();
55
56     catch (MySqlException ex)
57     {
58         MessageBox.Show(ex.Message);
59     }
60     finally
61     {
62         conexionMySQL.closeConexion();
63     }
64
65     return mProductos;
66 }
67
68 1 referencia
69 public bool agregarProducto(Producto mProducto)
70 {
71     string INSERT = "INSERT INTO producto (nombre, precio, cantidad, imagen)" + " values (@nombre, @precio, @cantidad, @imagen)";
72
73     MySqlCommand mCommand = new MySqlCommand(INSERT);
74     mCommand.Connection = conexionMySQL.getConexion();
75     mCommand.Parameters.AddWithValue("@nombre", mProducto.nombre);
76     mCommand.Parameters.AddWithValue("@precio", mProducto.precio);
77     mCommand.Parameters.AddWithValue("@cantidad", mProducto.cantidad);
78     mCommand.Parameters.AddWithValue("@imagen", mProducto.imagen);
79     mCommand.ExecuteNonQuery();
80 }

```

Ilustración 16 Clase ProductoConsulta 2



```

71
72 MySqlCommand mCommand = new MySqlCommand(INSERT, conexionMySQL.getConexion());
73 mCommand.Parameters.Add(new MySqlParameter("@nombre", mProducto.nombre));
74 mCommand.Parameters.Add(new MySqlParameter("@precio", mProducto.precio));
75 mCommand.Parameters.Add(new MySqlParameter("@cantidad", mProducto.cantidad));
76 mCommand.Parameters.Add(new MySqlParameter("@imagen", mProducto.imagen));
77
78 return mCommand.ExecuteNonQuery() > 0;
79
80
81 1 referencia
82 internal bool modificarProducto(Producto mProducto)
83 {
84     string UPDATE = " UPDATE producto " +
85     "SET nombre = @nombre, " +
86     "precio = @precio, " +
87     "cantidad = @cantidad, " +
88     "imagen = @imagen " +
89     "WHERE producto_id = @producto_id ";
90
91     MySqlCommand mCommand = new MySqlCommand(UPDATE, conexionMySQL.getConexion());
92     mCommand.Parameters.Add(new MySqlParameter("@nombre", mProducto.nombre));
93     mCommand.Parameters.Add(new MySqlParameter("@precio", mProducto.precio));
94     mCommand.Parameters.Add(new MySqlParameter("@cantidad", mProducto.cantidad));
95     mCommand.Parameters.Add(new MySqlParameter("@imagen", mProducto.imagen));
96     mCommand.Parameters.Add(new MySqlParameter("@producto_id", mProducto.producto_id));
97
98     return mCommand.ExecuteNonQuery() > 0;
99
100
101 1 referencia
102 internal bool eliminarProducto(Producto mProducto)
103 {
104     string DELETE = " DELETE FROM producto WHERE producto_id =@producto_id ";
105     MySqlCommand mCommand = new MySqlCommand(DELETE, conexionMySQL.getConexion());
106     mCommand.Parameters.Add(new MySqlParameter("@producto_id", mProducto.producto_id));
107     return mCommand.ExecuteNonQuery() > 0;
108
109 }

```

Ilustración 17 Clase ProductoConsulta 3

```

1 using System;
2 using System.Collections.Generic;
3 using System.Drawing.Imaging;
4 using System.Drawing;
5 using System.IO;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9
10 namespace projfinal
11 {
12     1 referencia
13     internal class Commun
14     {
15         public static int ROW_HEIGHT = 40;
16         0 referencias
17         public byte[] ImageToByteArray(Image mImage)
18         {
19             if (mImage == null)
20                 return null;
21
22             MemoryStream mMemoryStream = new MemoryStream();
23             mImage.Save(mMemoryStream, ImageFormat.Png);
24             return mMemoryStream.ToArray();
25
26         0 referencias
27         public Image BitmapToImage(Bitmap bitmap)
28         {
29             MemoryStream memoryStream = new MemoryStream();
30             bitmap.Save(memoryStream, ImageFormat.Png);
31             return Image.FromStream(memoryStream);
32
33         }
34     }
35 }

```

Ilustración 18 Clase Commun (Imagen)

```

1  using MySql.Data.MySqlClient;
2  using System;
3  using System.Collections.Generic;
4  using System.ComponentModel;
5  using System.Data;
6  using System.Drawing;
7  using System.Linq;
8  using System.Text;
9  using System.Threading.Tasks;
10 using System.Windows.Forms;
11
12 namespace projfinal
13 {
14     5 referencias
15     public partial class Form1 : Form
16     {
17         2 referencias
18         public Form1()
19         {
20             InitializeComponent();
21         }
22
23         1 referencia
24         private void ingresar_Click(object sender, EventArgs e)
25         {
26             String email, password;
27             email = usuario.Text;
28             password = contraseña.Text;
29             MySqlConnection connection = new MySqlConnection("server=localhost; database=proyec_final; User id=root;");
30             try
31             {
32                 connection.Open();
33             }
34             catch (Exception ex)
35             {
36                 MessageBox.Show("Error" + ex.ToString());
37                 throw;
38             }
39             String sql = "select email, password from usuario2 where email =' " + email + "' AND password =' " + password + "'";
40             MySqlCommand cmd = new MySqlCommand(sql, connection);
41             MySqlDataReader read = cmd.ExecuteReader();
42             if (read.Read())
43             {

```

Ilustración 19 Código de form1 (Inicio)

```

42             if (read.Read())
43             {
44                 this.Hide();
45                 MessageBox.Show("Bienvenido " + email);
46                 Categoria cat = new Categoria();
47                 cat.Show();
48                 this.Hide();
49             }
50             else
51             {
52                 MessageBox.Show("Usuario o Contraseña Incorrecto");
53             }
54         }
55
56         1 referencia
57         private void usuario_Leave(object sender, EventArgs e)
58         {
59             if (usuario.Text == "")
60             {
61                 usuario.Text = "Ingresar Usuario";
62                 usuario.ForeColor = Color.Black;
63             }
64         }
65
66         1 referencia
67         private void usuario_Enter(object sender, EventArgs e)
68         {
69             if (usuario.Text == "Ingresar Usuario")
70             {
71                 usuario.Text = "";
72                 usuario.ForeColor = Color.Black;
73             }
74         }
75
76         1 referencia
77         private void contraseña_Leave(object sender, EventArgs e)
78         {
79             if (contraseña.Text == "")
80             {
81                 contraseña.Text = "Ingresar Contraseña";
82                 contraseña.ForeColor = Color.Black;
83                 contraseña.UseSystemPasswordChar = false;
84             }
85         }
86     }

```

Ilustración 20 Código de form1 (Inicio) 2



```

83     }
84
85     1 referencia
86     private void contraseña_Enter(object sender, EventArgs e)
87     {
88         if (contraseña.Text == "Ingresar Contraseña")
89         {
90             contraseña.Text = "";
91             contraseña.ForeColor = Color.Black;
92             contraseña.UseSystemPasswordChar = true;
93         }
94     }
95
96     1 referencia
97     private void mostrar_CheckedChanged(object sender, EventArgs e)
98     {
99         if (mostrar.Checked == true)
100         {
101             contraseña.UseSystemPasswordChar = false;
102         }
103         else
104         {
105             contraseña.UseSystemPasswordChar = true;
106         }
107     }
108
109     1 referencia
110     private void Registrarse_Click(object sender, EventArgs e)
111     {
112         Registro f2 = new Registro();
113         f2.Show();
114         this.Hide();
115     }
116
117     1 referencia
118     private void cerrar_Click(object sender, EventArgs e)
119     {
120         Application.Exit();
121     }
122
123     }
124
125     }
126
127     }
128
129     }
130
131     }
132
133     }
134
135     }
136
137     }
138
139     }
140
141     }
142
143     }
144
145     }
146
147     }
148
149     }
150
151     }
152
153     }
154
155     }
156
157     }
158
159     }
160
161     }
162
163     }
164
165     }
166
167     }
168
169     }
170
171     }
172
173     }
174
175     }
176
177     }
178
179     }
180
181     }
182
183     }
184
185     }
186
187     }
188
189     }
190
191     }
192
193     }
194
195     }
196
197     }
198
199     }
200
201     }
202
203     }
204
205     }
206
207     }
208
209     }
210
211     }
212
213     }
214
215     }
216
217     }
218
219     }
220
221     }
222
223     }
224
225     }
226
227     }
228
229     }
230
231     }
232
233     }
234
235     }
236
237     }
238
239     }
240
241     }
242
243     }
244
245     }
246
247     }
248
249     }
250
251     }
252
253     }
254
255     }
256
257     }
258
259     }
260
261     }
262
263     }
264
265     }
266
267     }
268
269     }
270
271     }
272
273     }
274
275     }
276
277     }
278
279     }
280
281     }
282
283     }
284
285     }
286
287     }
288
289     }
290
291     }
292
293     }
294
295     }
296
297     }
298
299     }
300
301     }
302
303     }
304
305     }
306
307     }
308
309     }
310
311     }
312
313     }
314
315     }
316
317     }
318
319     }
320
321     }
322
323     }
324
325     }
326
327     }
328
329     }
330
331     }
332
333     }
334
335     }
336
337     }
338
339     }
340
341     }
342
343     }
344
345     }
346
347     }
348
349     }
350
351     }
352
353     }
354
355     }
356
357     }
358
359     }
360
361     }
362
363     }
364
365     }
366
367     }
368
369     }
370
371     }
372
373     }
374
375     }
376
377     }
378
379     }
380
381     }
382
383     }
384
385     }
386
387     }
388
389     }
390
391     }
392
393     }
394
395     }
396
397     }
398
399     }
400
401     }
402
403     }
404
405     }
406
407     }
408
409     }
410
411     }
412
413     }
414
415     }
416
417     }
418
419     }
420
421     }
422
423     }
424
425     }
426
427     }
428
429     }
430
431     }
432
433     }
434
435     }
436
437     }
438
439     }
440
441     }
442
443     }
444
445     }
446
447     }
448
449     }
450
451     }
452
453     }
454
455     }
456
457     }
458
459     }
460
461     }
462
463     }
464
465     }
466
467     }
468
469     }
470
471     }
472
473     }
474
475     }
476
477     }
478
479     }
480
481     }
482
483     }
484
485     }
486
487     }
488
489     }
490
491     }
492
493     }
494
495     }
496
497     }
498
499     }
500
501     }
502
503     }
504
505     }
506
507     }
508
509     }
510
511     }
512
513     }
514
515     }
516
517     }
518
519     }
520
521     }
522
523     }
524
525     }
526
527     }
528
529     }
530
531     }
532
533     }
534
535     }
536
537     }
538
539     }
540
541     }
542
543     }
544
545     }
546
547     }
548
549     }
550
551     }
552
553     }
554
555     }
556
557     }
558
559     }
560
561     }
562
563     }
564
565     }
566
567     }
568
569     }
570
571     }
572
573     }
574
575     }
576
577     }
578
579     }
580
581     }
582
583     }
584
585     }
586
587     }
588
589     }
590
591     }
592
593     }
594
595     }
596
597     }
598
599     }
600
601     }
602
603     }
604
605     }
606
607     }
608
609     }
610
611     }
612
613     }
614
615     }
616
617     }
618
619     }
620
621     }
622
623     }
624
625     }
626
627     }
628
629     }
630
631     }
632
633     }
634
635     }
636
637     }
638
639     }
640
641     }
642
643     }
644
645     }
646
647     }
648
649     }
650
651     }
652
653     }
654
655     }
656
657     }
658
659     }
660
661     }
662
663     }
664
665     }
666
667     }
668
669     }
670
671     }
672
673     }
674
675     }
676
677     }
678
679     }
680
681     }
682
683     }
684
685     }
686
687     }
688
689     }
690
691     }
692
693     }
694
695     }
696
697     }
698
699     }
699

```

Ilustración 21 Código de form1 (Inicio) 3

```

1  using MySql.Data.MySqlClient;
2  using System;
3  using System.Windows.Forms;
4
5  namespace proffinal
6  {
7      1 referencia
8      public partial class Registro : Form
9      {
10         2 referencia
11         public Registro()
12         {
13             InitializeComponent();
14         }
15
16         1 referencia
17         private void guardar_Click(object sender, System.EventArgs e)
18         {
19             MySqlConnection connection = new MySqlConnection("server=localhost; database=proyec_final; User id=root;");
20
21             try
22             {
23                 connection.Open();
24             }
25             catch (Exception ex)
26             {
27                 MessageBox.Show("Error" + ex.ToString());
28                 throw;
29             }
30
31             String sql = "insert into usuario2(email,password) values ('" + usuario2.Text + "','" + contraseña2.Text + "')";
32             MySqlCommand cmd = new MySqlCommand(sql, connection);
33
34             try
35             {
36                 cmd.ExecuteNonQuery();
37                 MessageBox.Show("REGISTRADO");
38                 this.Hide();
39                 Form1 f1 = new Form1();
40                 f1.Show();
41             }
42             catch (MySqlException ex)
43             {
44                 MessageBox.Show("Error" + ex.ToString());
45             }
46         }
47
48         1 referencia
49         private void cerrar2_Click(object sender, EventArgs e)
50         {
51             Application.Exit();
52         }
53     }
54 }
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Ilustración 22 Código de Form2 Registro

```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace projfinal
12 {
13     // referencia
14     public partial class Categoria : Form
15     {
16         // referencia
17         public Categoria()
18         {
19             InitializeComponent();
20         }
21
22         // referencia
23         private void cerrar2_Click(object sender, EventArgs e)
24         {
25             Application.Exit();
26         }
27
28         // referencia
29         private void pictureBox1_Click(object sender, EventArgs e)
30         {
31             Inventario ini = new Inventario();
32             ini.Show();
33             this.Hide();
34         }
35
36         // referencia
37         private void pictureBox2_Click(object sender, EventArgs e)
38         {
39             Registro cat = new Registro();
40             cat.Show();
41             this.Hide();
42         }
43
44         // referencia
45         private void Categoria_Load(object sender, EventArgs e)
46         {
47         }
48     }
49 }
```

Ilustración 23 Código de Form3 Categoría

```

1  using MySQL.XDevAPI.Relational;
2  using System;
3  using System.Collections.Generic;
4  using System.ComponentModel;
5  using System.Data;
6  using System.Drawing;
7  using System.Drawing.Imaging;
8  using System.IO;
9  using System.Linq;
10 using System.Security.Cryptography.X509Certificates;
11 using System.Text;
12 using System.Threading.Tasks;
13 using System.Windows.Forms;
14 using static System.Net.WebRequestMethods;
15 using static System.Windows.Forms.VisualStyles.VisualStyleElement.ListView;
16
17 namespace proffinal
18 {
19     7 referencias
20     public partial class Inventario : Form
21     {
22         private List<Producto> mProductos;
23         private Producto mProducto;
24         private ProductoConsultas mProductoConsultas;
25         2 referencias
26         public Inventario()
27         {
28             InitializeComponent();
29             mProductos = new List<Producto>();
30             mProducto = new Producto();
31             mProductoConsultas = new ProductoConsultas();
32             cargarProductos();
33         }
34         5 referencias
35         private void cargarProductos(string filtro = "")
36         {
37             dgvProductos.Rows.Clear();
38             dgvProductos.Refresh();
39             mProductos.Clear();
40             mProductos = mProductoConsultas.getProductos(filtro);
41             for (int i = 0; i < mProductos.Count(); i++)
42             {
43                 dgvProductos.RowTemplate.Height = Commun.ROW_HEIGHT;
44                 dgvProductos.Rows.Add(
45                     mProductos[i].producto_id,
46                     mProductos[i].nombre,
47                     mProductos[i].precio,
48                     mProductos[i].cantidad,
49                     Image.FromStream(new MemoryStream(mProductos[i].imagen)));
50             }
51         }
52         1 referencia
53         private void btregreso_Click(object sender, EventArgs e)
54         {
55             Categoria cat = new Categoria();
56             cat.Show();
57             this.Hide();
58         }
59         1 referencia
60         private void textbuscar_TextChanged(object sender, EventArgs e)
61         {
62             cargarProductos(textbuscar.Text.Trim());
63         }
64         2 referencias
65         private bool datosCorrectos()
66         {
67             // Validar el nombre del producto.
68             if (textnombre.Text.Trim().Equals(""))
69             {
70                 MessageBox.Show("Ingrese el nombre del producto");
71                 return false;
72             }
73         }
74     }
75

```

Ilustración 24 Código de Form4 Inventario venta

```

39         mProductos.Clear();
40         mProductos = mProductoConsultas.getProductos(filtro);
41         for (int i = 0; i < mProductos.Count(); i++)
42         {
43             dgvProductos.RowTemplate.Height = Commun.ROW_HEIGHT;
44             dgvProductos.Rows.Add(
45                 mProductos[i].producto_id,
46                 mProductos[i].nombre,
47                 mProductos[i].precio,
48                 mProductos[i].cantidad,
49                 Image.FromStream(new MemoryStream(mProductos[i].imagen)));
50         }
51     }
52     1 referencia
53     private void btregreso_Click(object sender, EventArgs e)
54     {
55         Categoria cat = new Categoria();
56         cat.Show();
57         this.Hide();
58     }
59     1 referencia
60     private void textbuscar_TextChanged(object sender, EventArgs e)
61     {
62         cargarProductos(textbuscar.Text.Trim());
63     }
64     2 referencias
65     private bool datosCorrectos()
66     {
67         // Validar el nombre del producto.
68         if (textnombre.Text.Trim().Equals(""))
69         {
70             MessageBox.Show("Ingrese el nombre del producto");
71             return false;
72         }
73     }
74 }
75

```

Ilustración 25 Código de Form4 Inventario venta 2

```
76 // Validar el precio del producto
77 if (textprecio.Text.Trim().Equals(""))
78 {
79     MessageBox.Show("Ingrese el precio del producto");
80     return false;
81 }
82
83 if (!float.TryParse(textprecio.Text.Trim(), out float precio))
84 {
85     MessageBox.Show("Ingrese un precio correcto");
86     return false;
87 }
88
89 // Validar la cantidad del producto.
90 if (textcantidad.Text.Trim().Equals(""))
91 {
92     MessageBox.Show("Ingrese la cantidad");
93     return false;
94 }
95
96
97 if (!int.TryParse(textcantidad.Text.Trim(), out int cant))
98 {
99     MessageBox.Show("Ingrese una cantidad correcta");
100     return false;
101 }
102
103 return true;
104 }
105
106 1 referencia
107 private void btagregar_Click(object sender, EventArgs e)
108 {
109     if (!datosCorrectos())
110     {
111         return;
112     }
113 }
```

Ilustración 26 Código de Form4 Inventario venta 3

```

111         CargarDatosProducto();
112
113         if (mProductoConsultas.agregarProducto(mProducto))
114         {
115             MessageBox.Show("Producto agregado");
116             cargarProductos();
117             LimpiarDatosProducto();
118         }
119         else
120             MessageBox.Show("Error al agregar el producto");
121     }
122
123     4 referencias
124     private void LimpiarDatosProducto()
125     {
126         textid.Text = "";
127         textnombre.Text = "";
128         textprecio.Text = "";
129         textcantidad.Text = "";
130         pbImage.Image = projfinal.Properties.Resources.agregar_imagen;
131     }
132
133     3 referencias
134     private void CargarDatosProducto()
135     {
136         mProducto.producto_id = getFolioIfExists();
137         mProducto.nombre = textnombre.Text.Trim();
138         mProducto.precio = float.Parse(textprecio.Text.Trim());
139         mProducto.cantidad = int.Parse(textcantidad.Text.Trim());
140         mProducto.imagen = ImageToByteArray(pbImage.Image);
141     }
142
143     2 referencias
144     private int getFolioIfExists()
145     {
146         if (!textid.Text.Trim().Equals(""))
147         {
148             if (int.TryParse(textid.Text.Trim(), out int folio))
149                 return folio;
150             else
151                 return -1;
152         }
153     }

```

Ilustración 27 Código de Form4 Inventario venta 4

```

150         else
151             return -1;
152     }
153
154     1 referencia
155     private byte[] ImageToByteArray(Image image)
156     {
157         if (image == null)
158             return null;
159         MemoryStream mMemoryStream = new MemoryStream();
160         image.Save(mMemoryStream, ImageFormat.Png);
161         return mMemoryStream.ToArray();
162     }
163
164     1 referencia
165     private void pbImage_Click(object sender, EventArgs e)
166     {
167         OpenFileDialog openFileDialog = new OpenFileDialog();
168         openFileDialog.InitialDirectory = Environment.GetFolderPath(Environment.SpecialFolder.MyPictures);
169         if (openFileDialog.ShowDialog() == DialogResult.OK)
170         {
171             pbImage.ImageLocation = openFileDialog.FileName;
172         }
173     }
174
175     1 referencia
176     private void dgvProductos_CellClick(object sender, DataGridViewCellEventArgs e)
177     {
178         try {
179             DataGridViewRow fila = dgvProductos.Rows[e.RowIndex];
180             textid.Text = Convert.ToString(fila.Cells["Folio"].Value);
181             textnombre.Text = Convert.ToString(fila.Cells["Nombre"].Value);
182             textprecio.Text = Convert.ToString(fila.Cells["Precio"].Value);
183             textcantidad.Text = Convert.ToString(fila.Cells["Cantidad"].Value);
184
185             MemoryStream ms = new MemoryStream();
186             Bitmap img = (Bitmap)dgvProductos.CurrentRow.Cells[4].Value;
187             img.Save(ms, ImageFormat.Png);
188             pbImage.Image = Image.FromStream(ms);
189         }
190     }

```

Ilustración 28 Código de Form4 Inventario venta 5



```
187     }
188     catch { }
189 }
190
191 1 referencia
192 private void btnmodificar_Click(object sender, EventArgs e)
193 {
194     if (!datosCorrectos())
195         return;
196     CargarDatosProducto();
197     if (mProductoConsultas.modificarProducto(mProducto))
198     {
199         MessageBox.Show("Producto modificado");
200         cargarProductos();
201         LimpiarDatosProducto();
202     }
203     else
204         MessageBox.Show("Error al modificar el producto");
205 }
206
207 1 referencia
208 private void bteliminar_Click(object sender, EventArgs e)
209 {
210     if (getFolioIfExist() == -1)
211     {
212         return;
213     }
214     if (MessageBox.Show("¿desea eliminar el producto?", "Eliminar Producto",
215         MessageBoxButtons.YesNo) == DialogResult.Yes)
216     {
217         CargarDatosProducto();
218         if (mProductoConsultas.eliminarProducto(mProducto))
219         {
220             MessageBox.Show("Producto Eliminado");
221             cargarProductos();
222             LimpiarDatosProducto();
223         }
224     }
225 }
```

Ilustración 29 Código de Form4 Inventario venta 6

```

226     }
227
228
229     1 referencia
230     private void btLimpiar_Click(object sender, EventArgs e)
231     {
232         LimpiarDatosProducto();
233     }
234
235
236
237     String id2, nombre;
238     decimal precio;
239     int cantidad;
240
241     float subtotal;
242
243     1 referencia
244     private void btAgregarproducto_Click(object sender, EventArgs e)
245     {
246
247         id2 = textid.Text;
248
249         nombre = textnombre.Text;
250
251         precio = Convert.ToDecimal(textprecio.Text);
252         cantidad = Convert.ToInt32(nudcantidadp.Text);
253
254         subtotal = (float)(precio * cantidad);
255
256         dataGridView1.Rows.Add(id2, nombre, precio, cantidad, subtotal);
257
258     }
259
260
261     1 referencia
262     private void dataGridView1_CellClick(object sender, DataGridViewCellEventArgs e)
263     {
264

```

Ilustración 30 Código de Form4 Inventario venta 7

```

265
266     try
267     {
268
269         id2 = dataGridView1.CurrentRow.Cells["Folio"].Value.ToString();
270         nombre = dataGridView1.CurrentRow.Cells["Nombre"].Value.ToString();
271         precio = (decimal)dataGridView1.CurrentRow.Cells["Precio"].Value;
272         cantidad = (int)dataGridView1.CurrentRow.Cells["Cantidad"].Value;
273         subtotal = (float)dataGridView1.CurrentRow.Cells["SubTotal"].Value;
274
275     }
276     catch
277     {
278
279     }
280
281
282
283
284
285
286
287     1 referencia
288     private void panel6_Paint(object sender, PaintEventArgs e)
289     {
290         cototal();
291         efectivo();
292     }
293
294     1 referencia
295     public void cototal()
296     {
297
298         decimal total = 0;
299
300         foreach (DataGridViewRow row in dataGridView1.Rows)
301         {
302             total += Convert.ToDecimal(row.Cells["Column5"].Value);
303         }

```

Ilustración 31 Código de Form4 Inventario venta 8

```

303         tpagar.Text = Convert.ToString(total);
304     }
305     decimal pag;
306     decimal efec;
307
308     1 referencia
309     public void efectivo()
310     {
311         try
312         {
313             pag = Convert.ToDecimal(tpagar.Text);
314             efec = Convert.ToDecimal(textefectivo.Text);
315
316         }
317         catch
318         {
319             textcambio.Text = Convert.ToString(efec - pag);
320         }
321     }
322
323     1 referencia
324     private void btppagar_Click(object sender, EventArgs e)
325     {
326         this.Hide();
327         MessageBox.Show("Gracias por tu compra ");
328
329         Inventario inv = new Inventario();
330         inv.Show();
331         this.Hide();
332     }
333
334
335
336
337
338
339

```

*Ilustración 32 Código de Form4 Inventario venta 9*

```

339
340     1 referencia
341     private void Inventario_Load(object sender, EventArgs e)
342     {
343     }
344
345     1 referencia
346     private void bteliminar_Click(object sender, EventArgs e)
347     {
348         dataGridView1.Rows.Remove(dataGridView1.CurrentRow);
349     }
350
351     1 referencia
352     private void dataGridView1_CellPainting(object sender, DataGridViewCellPaintingEventArgs e)
353     {
354     }
355
356
357
358
359
360

```

*Ilustración 33 Código de Form4 Inventario venta 10*



## PRUEBAS MANUALES



Ilustración 34 inicio

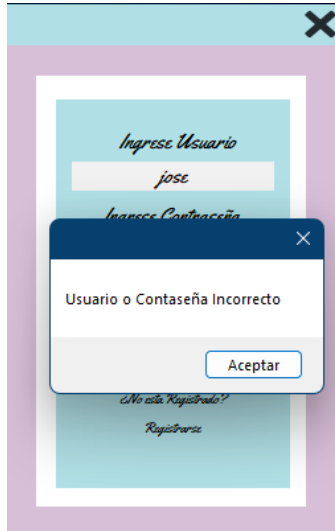


Ilustración 35 mensaje de error inicio



Ilustración 36 registro

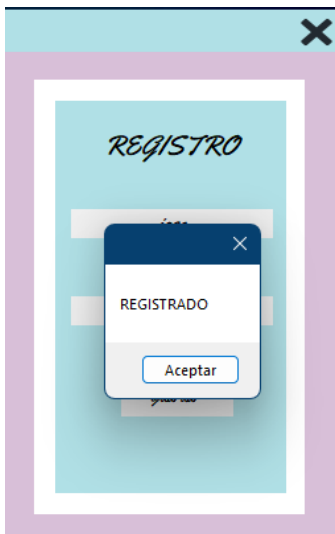


Ilustración 38 Registro guardado



Ilustración 37 Ingresar Inicio

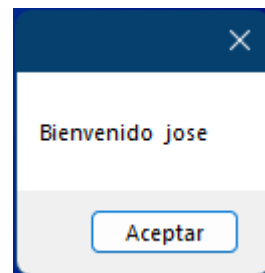


Ilustración 39 Mensaje de bienvenida

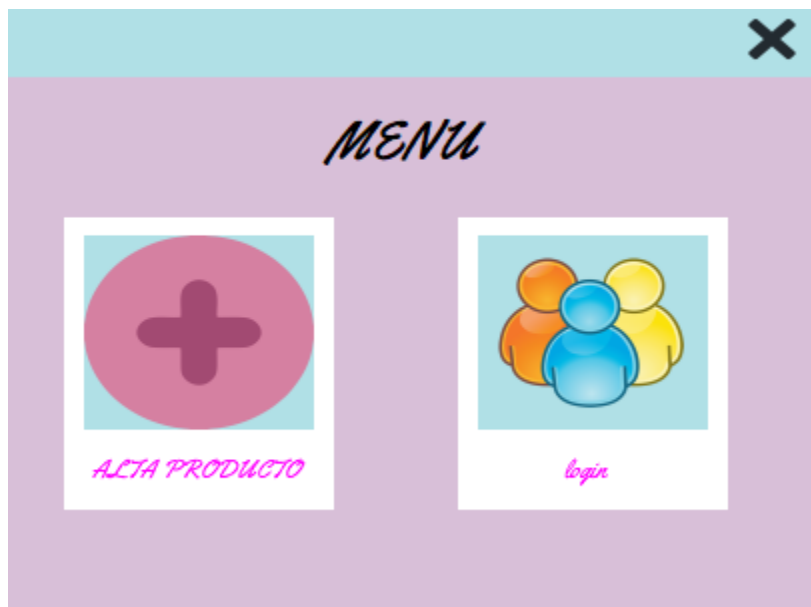


Ilustración 40 Menú

### ALTA DE PRODUCTOS

ID:

NOMBRE:

PRECIO:

CANTIDAD:

Folio	Nombre	Precio	Cantidad	Imagen
9	pepsi	8	23	
12	coca	5	34	

IMAGEN

### VENTA PRODUCTOS

Cantidad Producto:

Folio	Nombre	Precio	CantidadP	Sub Total

Total Pagar \$:

Efectivo:

Cambio:

Ilustración 41 Alta de Producto

### ALTA DE PRODUCTOS

ID:

NOMBRE:   IMAGEN

PRECIO:

CANTIDAD:

Folio	Nombre	Precio	Cantidad	Imagen
9	pepsi	8	23	
12	coca	5	34	

### VENTA PRODUCTOS

Contidud Producto:

Folio	Nombre	Precio	CantidadP	SubTotal

Total Pagar \$: 
Cambio:

Producto agregado

Ilustración 42 Producto guardado

### ALTA DE PRODUCTOS

ID:

NOMBRE:   IMAGEN

PRECIO:

CANTIDAD:

Folio	Nombre	Precio	Cantidad	Imagen
12	coca	5	34	
18	laptop	1500	21	

### VENTA PRODUCTOS

Contidud Producto:

Folio	Nombre	Precio	CantidadP	SubTotal

Total Pagar \$: 
Cambio:

Ilustración 43 Producto registrado en BD

### ALTA DE PRODUCTOS

ID:

NOMBRE:   IMAGEN

PRECIO:

CANTIDAD:

Folio	Nombre	Precio	Cantidad	Imagen
12	coca	5	34	
18	laptop	1500	21	

### VENTA PRODUCTOS

Contidud Producto:

Folio	Nombre	Precio	CantidadP	SubTotal
18	laptop	1500	3	4500

Total Pagar \$: 
Cambio:

Ilustración 44 Venta de Producto

### ALTA DE PRODUCTOS

ID:

NOMBRE:

PRECIO:

CANTIDAD:

Folio	Nombre	Precio	Cantidad	Imagen
12	coca	5	34	
18	laptop	1500	21	

### VENTA PRODUCTOS

Cantidad Producto:

Folio	Nombre	Precio	CantidadP	SubTotal
18	laptop	1500	3	4500

Total Pagar \$: 4500

Efectivo:

Cambio:

Ilustración 45 Pago Efectivo

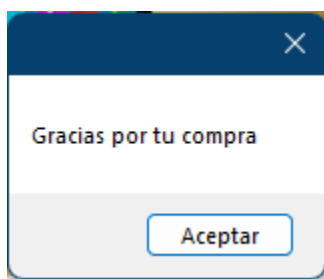


Ilustración 46 Mensaje de éxito de Cobro

## COLUSIÓN

El código representa una aplicación de inventario y gestión de productos en el contexto de una tienda. La aplicación utiliza Windows Forms y una base de datos MySQL para almacenar la información de los productos y los usuarios, la aplicación consta de tres formularios principales: "Form1" para el inicio de sesión de los usuarios, "Categoria" para seleccionar la categoría de productos e "Inventario" para administrar los productos y realizar ventas.

El formulario "Inventario" permite al usuario agregar, modificar y eliminar productos, así como realizar ventas. Los productos se muestran en un DataGridView y se cargan desde la base de datos. También se puede cargar una imagen para cada producto. Además, se pueden agregar productos a una lista de compras, donde se muestra su información, y se puede realizar el cálculo del total a pagar y el cambio al recibir el pago, el código implementa una aplicación de gestión de inventario y ventas básica utilizando Windows Forms y una base de datos MySQL. Esta



aplicación permite a los usuarios administrar productos, realizar ventas y mantener un seguimiento de las transacciones. Sin embargo, el código también muestra algunas áreas de mejora potencial, como la separación de la lógica de presentación y acceso a la base de datos, así como la implementación de una mejor validación de datos y gestión de errores.

