



INSTITUTO TECNOLÓGICO DE TIJUANA



DEPARTAMENTO
DE SISTEMAS
Y COMPUTACION

Carrera:



INGENIERÍA EN
SISTEMAS
COMPUTACIONALES

Nombre o Número del Equipo:

Rosales Gómez Valeria

Materia:

Patrones de diseño

Periodo:

Agosto - Diciembre 2025

Unidad y Tema:

Examen Unidad 4 y 5

Fecha: Tijuana, Baja California a Diciembre 11 del 2025



Código

The image displays three vertically stacked screenshots of the Microsoft Visual Studio IDE interface, showing code editors for different files within a single solution named "ExamenU4-5".

Screenshot 1: The code editor shows the file `Material.cs` from the `ExamenU4-5` project. The code defines a class `Material` with properties `Nombre`, `EsReciclable`, and `Estado`, and a constructor `Iniciar()`.

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using ExamenU4_5.Negocio;
7
8 namespace ExamenU4_5.Modelo
9 {
10     public class Material
11     {
12         public string Nombre { get; set; }
13         public bool EsReciclable { get; set; }
14         public IEstadoMaterial Estado { get; set; }
15
16         public void Iniciar()
17         {
18             Nombre = string.Empty;
19             EsReciclable = false;
20             Estado = null;
21         }
22     }
23 }
```

Screenshot 2: The code editor shows the file `PoolMateriales.cs`. It defines a class `PoolMateriales` with a private stack `_materialesDisponibles` and methods `Obtener()` and `Devolver(Material material)`.

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using ExamenU4_5.Modelo;
7
8 namespace ExamenU4_5.Negocio
9 {
10     public class PoolMateriales
11     {
12         private readonly Stack<Material> _materialesDisponibles = new Stack<Material>();
13
14         public Material Obtener()
15         {
16             if (_materialesDisponibles.Count > 0)
17             {
18                 return _materialesDisponibles.Pop();
19             }
20             else
21             {
22                 return new Material();
23             }
24         }
25
26         public void Devolver(Material material)
27         {
28             material.Iniciar();
29             _materialesDisponibles.Push(material);
30         }
31     }
32 }
```

Screenshot 3: The code editor shows the file `ITraductor.cs`. It defines an interface `ITraductor` with a method `ObtenerTraducción()`.

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace ExamenU4_5.Negocio
8 {
9     public interface ITraductor
10     {
11         string ObtenerTraducción();
12     }
13 }
```

The Solution Explorer on the right side of each screenshot lists the files in the solution, including `ExamenU4-5`, `Properties`, `References`, `Modelo` (containing `Material.cs`), `Negocio` (containing `AdaptadorIngles.cs`, `ControlMaestroProducción.cs`, `EstadoNoReciclable.cs`, `EstadoReciclable.cs`, `EstadoMaterial.cs`, `ITraductor.cs`, `PoolMateriales.cs`, and `Program.cs`), `Presentación`, and `App.config`.



ExamenU4-5

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;

6
7  namespace ExamenU4_5.Negocio
8  {
9      public interface IEstadoMaterial
10     {
11         string ObtenerEstado();
12         bool EsRecicitable();
13     }
14 }
15 
```

ExamenU4-5

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;

6
7  namespace ExamenU4_5.Negocio
8  {
9      public class EstadoReciclable : IEstadoMaterial
10     {
11         public string ObtenerEstado() => "Reciclable";
12         public bool EsRecicitable() => true;
13     }
14 }
15 
```

ExamenU4-5

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;

6
7  namespace ExamenU4_5.Negocio
8  {
9      public class EstadoNoReciclable : IEstadoMaterial
10     {
11         public string ObtenerEstado() => "No reciclable";
12         public bool EsRecicitable() => false;
13     }
14 }
15 
```

Explorador de soluciones

Buscar en Explorador de soluciones (Ctrl+F)

- Solución "ExamenU4-5" (1 de 1 proyecto)
 - Properties
 - Referencias
 - Modelo
 - Material.cs
 - Negocio
 - AdaptadorIngles.cs
 - ControlMaestroProducción.cs
 - EstadoNoReciclable.cs
 - EstadoReciclable.cs
 - IEstadoMaterial.cs
 - ITraductor.cs
 - PoolMateriales.cs
 - Presentación
 - App.config

Explorador de soluciones

Buscar en Explorador de soluciones (Ctrl+F)

- Solución "ExamenU4-5" (1 de 1 proyecto)
 - Properties
 - Referencias
 - Modelo
 - Material.cs
 - Negocio
 - AdaptadorIngles.cs
 - ControlMaestroProducción.cs
 - EstadoNoReciclable.cs
 - EstadoReciclable.cs
 - IEstadoMaterial.cs
 - ITraductor.cs
 - PoolMateriales.cs
 - Presentación
 - App.config

Explorador de soluciones

Buscar en Explorador de soluciones (Ctrl+F)

- Solución "ExamenU4-5" (1 de 1 proyecto)
 - Properties
 - Referencias
 - Modelo
 - Material.cs
 - Negocio
 - AdaptadorIngles.cs
 - ControlMaestroProducción.cs
 - EstadoNoReciclable.cs
 - EstadoReciclable.cs
 - IEstadoMaterial.cs
 - ITraductor.cs
 - PoolMateriales.cs
 - Presentación
 - App.config



Educación

Secretaría de Educación Pública



The screenshot shows the Microsoft Visual Studio interface. The code editor on the left displays the C# file `ControlMaestroProduccion.cs` from the project `ExamenU4_5.Negocio`. The code implements a static factory pattern to manage recyclable materials. The Solution Explorer on the right shows the project structure, including files like `AdaptadorIngresos.cs`, `EstadoNoReciclable.cs`, `EstadoReciclable.cs`, `EstadoMaterial.cs`, `ITraductor.cs`, `PoolMateriales.cs`, `Program.cs`, and `Material.cs`.

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace ExamenU4_5.Negocio
8  {
9      public class ControlMaestroProduccion
10     {
11         private static ControlMaestroProduccion _instancia;
12         private HashSet<string> materialesReciclables;
13
14         private ControlMaestroProduccion()
15         {
16             materialesReciclables = new HashSet<string>(StringComparer.OrdinalIgnoreCase)
17             {
18                 "plastico", "Papel", "Vidrio", "Metal", "Carton", "Pilas", "Latas"
19             };
20         }
21
22
23         public static ControlMaestroProduccion Instancia
24         {
25             get
26             {
27                 if (_instancia == null)
28                     _instancia = new ControlMaestroProduccion();
29                 return _instancia;
30             }
31         }
32
33         public bool EsReciclable(string nombre)
34         {
35             return materialesReciclables.Contains(nombre);
36         }
37     }
38 }
```

The screenshot shows the Visual Studio IDE interface. The code editor on the left displays the file `ExamenU4_5.cs` with the following content:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using ExamenU4_5.Modelo;
7
8  namespace ExamenU4_5.Negocio
9  {
10     public class AdaptadorIngles : ITraductor
11     {
12         private Material _material;
13
14         public AdaptadorIngles(Material material)
15         {
16             _material = material;
17         }
18
19         public string ObtenerTraduccion()
20         {
21             string texto = _material.Nombre;
22
23             texto = texto.Replace("Plastico", "Plastic")
24                         .Replace("Papel", "Paper")
25                         .Replace("Vidrio", "Glass")
26                         .Replace("Carton", "Cardboard")
27                         .Replace("Pilas", "Batteries")
28                         .Replace("Latas", "Cans")
29                         .Replace("Tela", "Fabric");
30
31             return texto;
32         }
33     }
34 }
```

The Solution Explorer on the right shows the project structure for `ExamenU4_5`, which includes files like `ControlMaestroProduccion.cs`, `EstadoNoReciclable.cs`, `EstadoReciclable.cs`, `IEstadoMaterial.cs`, `ITraductor.cs`, `PoolMateriales.cs`, `Program.cs`, `Material.cs`, and `App.config`.

The screenshot shows the Microsoft Visual Studio IDE interface. The code editor on the left displays the `Program.cs` file for the `ExamenU4_5` project. The code implements a factory pattern to simulate industrial processes. The Solution Explorer on the right shows the project structure, including files like `AdaptadorIngles.cs`, `ControlMaestroProduccion.cs`, `EstadoNoRecicitable.cs`, `EstadoRecicitable.cs`, `EstadoMaterial.cs`, `ITraductor.cs`, `PoolMateriales.cs`, `Program.cs`, `Materiales.cs`, and `App.config`.

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using ExamenU4_5.Modelo;
7  using ExamenU4_5.Negocio;
8
9  namespace ExamenU4_5.Presentacion
10 {
11     class Program
12     {
13         static void Main(string[] args)
14         {
15             Console.WriteLine("Simulación de procesos industriales automatizados - Materiales Reciclab");
16
17             var pool = new PoolMateriales();
18             var control = ControlMaestroProduccion.Instancia;
19
20             var listaMateriales = new List<string>
21             {
22                 "Plastico", "Papel", "Tela", "Metal", "Carton"
23             };
24
25             var materialesProcesados = new List<Material>();
26
27             foreach (var nombre in listaMateriales)
28             {
29                 var material = pool.Obtener();
30                 material.Nombre = nombre;
31                 material.EsRecicitable = control.EsRecicitable(nombre);
32
33                 if (material.EsRecicitable)
34                     material.Estado = new EstadoRecicitable();
35                 else
36                     material.Estado = new EstadoNoRecicitable();
37
38             }
39
40             foreach (var material in materialesProcesados)
41             {
42                 Console.WriteLine(material.Nombre + " " + material.EsRecicitable);
43             }
44
45         }
46     }
47 }
```



The screenshot shows the Microsoft Visual Studio IDE interface. The code editor displays the file `ExamenU4-5.cs` with the following content:

```
AdaptadorIngles.cs ControlMaestroProduccion.cs EstadoNoRecicable.cs EstadoRecicitable.cs IEstadoMaterial.cs ITraductor.cs PoolMateriales.cs Program.cs Material.cs
[ExamenU4-5]
34     material.Estado = new EstadoRecicitable();
35   else
36     material.Estado = new EstadoNoRecicitable();
37 
38   materialesProcesados.Add(material);
39 }
40 
41 Console.WriteLine("Materiales presentados");
42 foreach (var m in materialesProcesados)
43 {
44   Console.WriteLine($"- {m.Nombre}");
45 }
46 
47 Console.WriteLine("Estado de los Materiales");
48 foreach (var m in materialesProcesados)
49 {
50   Console.WriteLine($"- {m.Nombre}: {m.Estado.ObtenerEstado()}");
51 }
52 
53 Console.WriteLine("State of Materials");
54 foreach (var m in materialesProcesados)
55 {
56   ITraductor traductor = new AdaptadorIngles(m);
57   string nombreEnIngles = traductor.ObtenerTraduccion();
58 
59   string estadoEnIngles = m.Estado.EsRecicitable() ? "Recyclable" : "Non-recyclable";
60 
61   Console.WriteLine($"- {nombreEnIngles}: {estadoEnIngles}");
62 }
63 
64 Console.WriteLine("");
65 Console.WriteLine("Proceso finalizado");
66 Console.ReadKey();
67 }
```

The Solution Explorer on the right shows the project structure for "ExamenU4-5" containing files like `AdaptadorIngles.cs`, `ControlMaestroProduccion.cs`, etc.

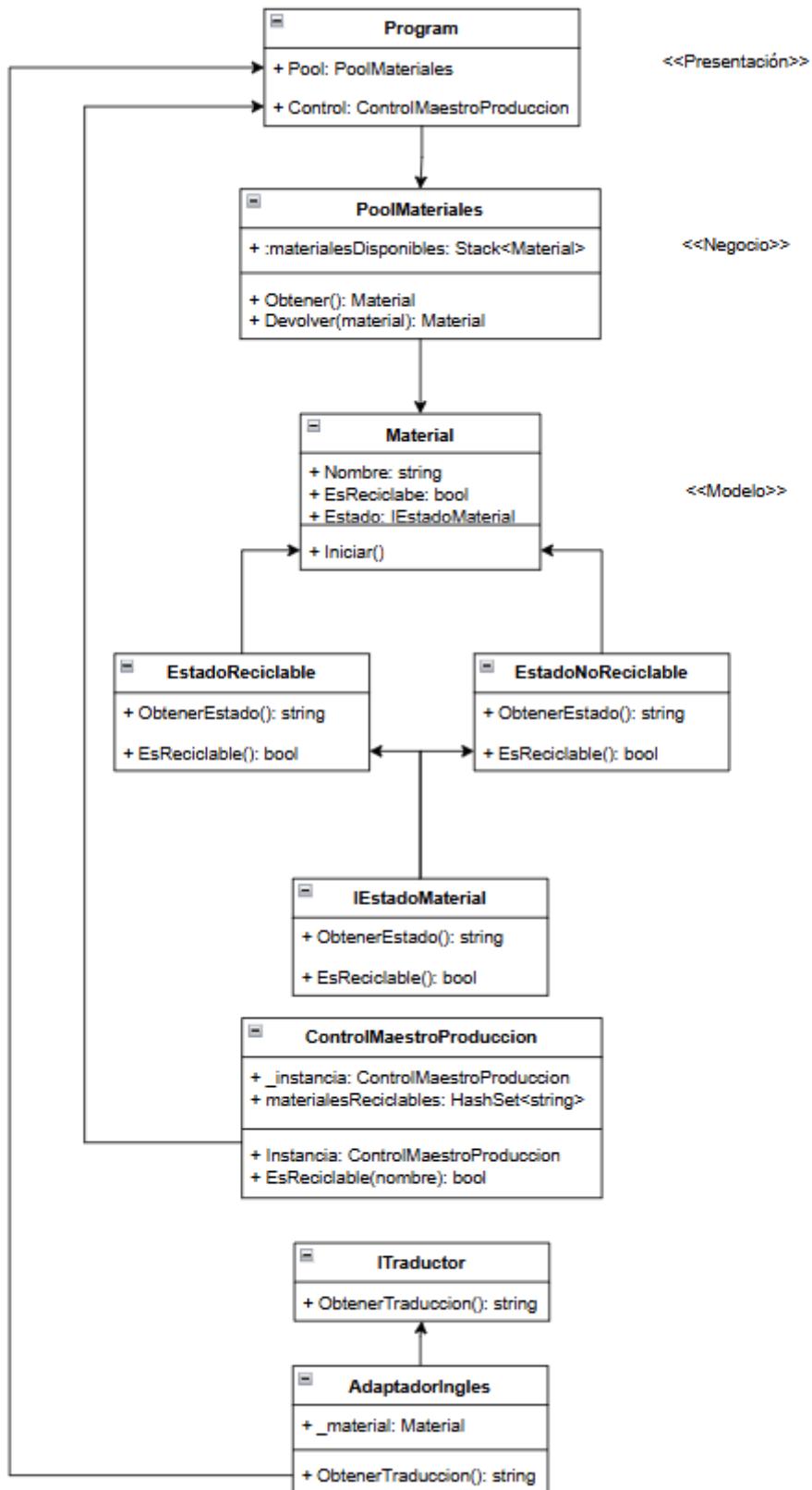


Resultados de pantalla

```
C:\Users\valer\source\repos\f  X  +  ▾  
Simulación de procesos industriales automatizados - Materiales Reciclables  
Materiales presentados  
- Plastico  
- Papel  
- Tela  
- Metal  
- Carton  
  
Estado de los Materiales  
- Plastico: Reciclable  
- Papel: Reciclable  
- Tela: No reciclable  
- Metal: Reciclable  
- Carton: Reciclable  
  
State of Materials  
- Plastic: Recyclable  
- Paper: Recyclable  
- Fabric: Non-recyclable  
- Metal: Recyclable  
- Cardboard: Recyclable  
  
Proceso finalizado
```



Diagrama de flujo





Conclusiones

Como resultado de los conocimientos adquiridos en la materia, se presentó un programa con 5 patrones de diseño implementados. A diferencia de programas realizados en semestres anteriores se puede notar una gran diferencia en la organización de la estructura de un código con patrones implementados y se concluye con la mención de la importancia de aplicarlos a futuro en programas con el fin de lograr proyectos más eficientes, organizados y mantenibles.