



Actividad | #1 |

Creación de Expediente

Lenguajes de Programación III

Ingeniería en Desarrollo de Software



TUTOR: Miguel Ángel Rodríguez Vega

ALUMNO: Oscar Esteban Sánchez Leyva

FECHA: 25/Noviembre/2025

ÍNDICE

ÍNDICE	2
DESARROLLO	3
CAPTURA DE LENGUAJE C#	3
VENTANA DE CONSOLA CON LOS DATOS GENERALES	5
VENTANA DE CONSOLA CON DATOS CAPTURADOS.....	7
APLICACIÓN DE ESCRITORIO CON .NET	9
CAPTURAS DE CADA REQUISITO SOLICITADO.....	14
REFERENCIAS.....	15

DESARROLLO

CAPTURA DE LENGUAJE C#

En el lenguaje de programación C#, la adquisición de datos se lleva a cabo predominantemente mediante el uso del método `Console.ReadLine()`, que permite leer la entrada proporcionada por el usuario a través de la consola y almacenarla como una variable de tipo `string`.







En el caso del desarrollo de interfaces gráficas, se emplean elementos como cuadros de diálogo, por ejemplo, `InputBox` (el cual requiere la inclusión de una referencia específica a una biblioteca DLL) o clases diseñadas para gestionar controles en formularios.

Adicionalmente, otra forma frecuente de captura es la realización de capturas de pantalla (screenshots), lo que se consigue utilizando métodos que renderizan el contenido visual mostrado en pantalla sobre un objeto de tipo `Bitmap`.

Visual Studio 2022

Abrir recientes

Anterior

	Database1.sln	20/09/2025 11:05 p. m.
	C:\Users\Lenovo\OneDrive\Documents\Universidad Coppel\Database1	
	Universidad México Int...	10/08/2025 12:17 p. m.
	C:\...OneDrive\Documents\Universidad Coppel\Universidad México Internacional	
	CentrosTrabajoApp.csproj	10/08/2025 10:08 a. m.
	C:\...Temp\6f995fac-88ac-4eea-a85e-a9dd722e89ee_CentrosTrabajoApp.zip.9ee	
	Univerdad Coppel.sln	05/08/2025 11:27 p. m.
	C:\Users\Lenovo\OneDrive\Documents\Universidad Coppel\Univerdad Coppel	
	COPPEL.sln	05/08/2025 11:17 p. m.
	C:\Users\Lenovo\OneDrive\Documents\Coppel\COPPEL	
	Coppel.sln	05/08/2025 11:08 p. m.
	C:\Users\Lenovo\OneDrive\Documents\Universidad Coppel\Coppel	

Comenzar



Clonar un repositorio

Obtiene código desde un repositorio en línea, como GitHub o Azure DevOps.



Abrir un proyecto o una solución

Abre un archivo .sln o proyecto de Visual Studio local.



Abrir una carpeta local

Navegar y editar el código en cualquier carpeta



Crear un proyecto

Elija una plantilla de proyecto con la técnica scaffolding de código para comenzar.

[Continuar sin código →](#)

Crear un proyecto

Buscar plantillas (Alt+S)

Borrar todo

Plantillas de proyecto recientes

- Proyecto de base de datos de SQL Server Lenguaje de consulta
- Aplicación de Windows Forms (.NET Framework) C#
- Aplicación de WPF (.NET Framework) C#
- Aplicación de escritorio de Windows C++
- Aplicación de consola C++
- Proyecto de CMake C++
- Proyecto vacío C++

C#

Todas las plataformas

Consola



Aplicación de consola

Proyecto para crear una aplicación de línea de comandos que se puede ejecutar en .NET en Windows, Linux y macOS.

C#

Linux

macOS

Windows

Consola



Aplicación de consola (.NET Framework)

Proyecto para crear una aplicación de línea de comandos.

C#

Windows

Consola

¿No encuentra lo que busca?
Instalar más herramientas y características

Configure su nuevo proyecto

Aplicación de consola (.NET Framework)

C#

Windows

Consola

Nombre del proyecto

Bienvenidos a UMI

Ubicación

C:\Users\Lenovo\OneDrive\Documents\Universidad Coppel

Nombre de la solución ⓘ

Bienvenidos a UMI

☐ Colocar la solución y el proyecto en el mismo directorio

Marco

.NET Framework 4.7.2

Proyecto se creará en "C:\Users\Lenovo\OneDrive\Documents\Universidad Coppel\Bienvenidos a UMI\Bienvenidos a UMI\"

VENTANA DE CONSOLA CON LOS DATOS GENERALES

Es necesario incluir una ilustración donde se observe la interfaz de la consola al momento de iniciar el programa. En dicha captura deben visualizarse los siguientes elementos:

- El título o presentación que identifica el sistema desarrollado.
- Un conjunto de datos básicos o un conjunto de instrucciones iniciales destinadas a orientar al usuario.
- Información preliminar que se despliega antes de proceder con la recopilación de datos por parte del usuario.

Este elemento funcional tiene como objetivo evidenciar que el programa inicie de manera adecuada y presenta al usuario la información general necesaria para su correcta utilización desde el inicio.

La finalidad principal de este requisito radica en comprobar que el programa se ejecuta de manera adecuada y que, desde el inicio, proporciona información precisa y comprensible al usuario antes de solicitar cualquier tipo de dato.

La captura de pantalla requerida actúa como evidencia que valida que el sistema cumple con los siguientes aspectos fundamentales:

- Está operativo.
- Se identifica correctamente.
- Ofrece instrucciones claras.
- Demuestra estar preparado para interactuar con el usuario.

```
Program.cs* X
Bienvenidos a UMI Bienvenidos_a_UMI.Program Main(string[] args)
5 using System.Threading.Tasks;
6
7 using System;
8 namespace Bienvenidos_a_UMI
9 {
10     0 referencias
11     internal class Program
12     {
13         0 referencias
14         static void Main(string[] args)
15         {
16             // Universidad México Internacional //
17             string Nombre;
18             string Edad;
19             string FechadeNacimiento;
20             string Carrera;
21             string Cierre;
22
23             Console.WriteLine("Bienvenidos a UMI/Universidad Coppel");
24             Console.WriteLine("\nIngresa los datos solicitados: ");
25             Console.WriteLine("\nNombre completo: ");
26             Nombre = Console.ReadLine();
27             Console.WriteLine("\nEdad: ");
28             Edad = Console.ReadLine();
29             Console.WriteLine("\nFecha de Nacimiento (DD/MM/AAAA): ");
30             FechadeNacimiento = Console.ReadLine();
31             Console.WriteLine("\nCarrera que desea ingresar: ");
32             Carrera = Console.ReadLine();
33
34             Console.WriteLine("\n\nGracia " + Nombre + "por formar parte de UMI/Universidad Coppel. \n Bienvenido a la carrera de " + Carrera + ".");
35             Console.WriteLine("\nNombre: " + Nombre);
36             Console.WriteLine("Edad: " + Edad);
37             Console.WriteLine("Fecha de Nacimiento: " + FechadeNacimiento);
38             Console.WriteLine("Carrera Asignada: " + Carrera);
39             Console.WriteLine("\n\n\n\n");
40             Console.ReadLine();
41         }
42     }
43 }
```

VENTANA DE CONSOLA CON DATOS CAPTURADOS

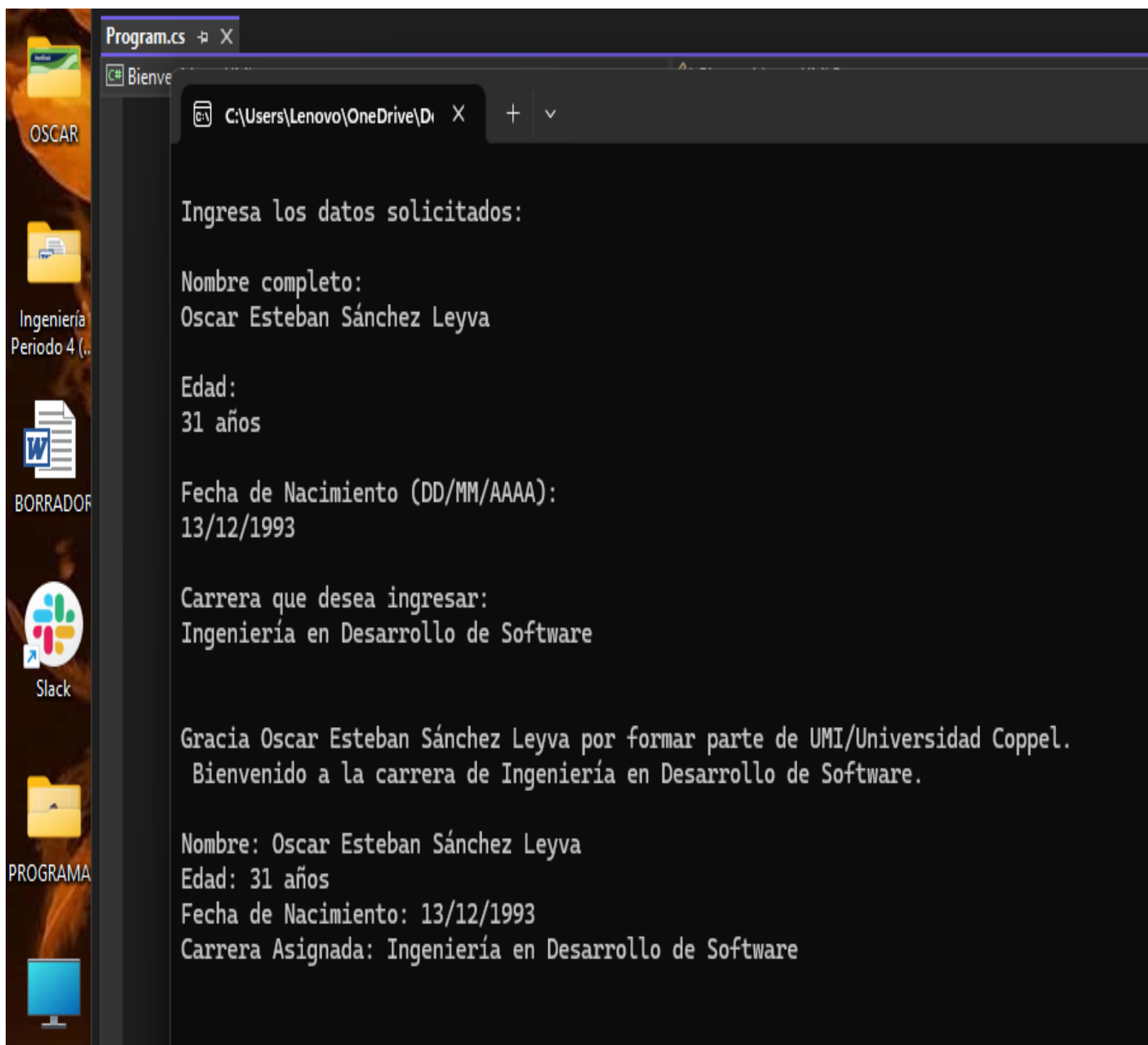
Se debe incluir una imagen que muestre el funcionamiento completo del programa en la consola. La captura debe evidenciar lo siguiente:

- El programa solicitó datos al usuario mediante la utilización de comandos de entrada (inputs).
- Los datos fueron ingresados correctamente por el usuario.
- Posteriormente, se presentaron los resultados, cálculos, confirmaciones o mensajes finales generados por el programa.

Esta imagen servirá como prueba del correcto funcionamiento del sistema, reflejando el proceso completo, desde la recepción de datos iniciales hasta la presentación de los resultados derivados.

El propósito central de esta información radica en validar y evidenciar que el programa opera de manera correcta y completa, utilizando una captura de pantalla como elemento probatorio. Dicha captura debe ilustrar de manera exhaustiva todo el flujo operativo del sistema a través de la consola. Esto incluye, en primer lugar, que el programa solicita datos al usuario mediante comandos de entrada; en segundo lugar, que el usuario proporciona dicha información como respuesta; y, finalmente, que el programa lleva a cabo el procesamiento correspondiente de los datos ingresados y presenta resultados, cálculos o mensajes finales.

A través de este procedimiento visual y verificable, se busca confirmar que el sistema no solo es capaz de iniciar, sino que además interactúa de forma adecuada con el usuario y genera una salida coherente basada en la información proporcionada. En síntesis, la captura de pantalla cumple la función de evidencia tangible que demuestra el funcionamiento integral del software, destacando cada etapa clave del proceso operativo.



APLICACIÓN DE ESCRITORIO CON .NET

Este apartado aborda la necesidad de evidenciar la creación de una aplicación equipada con una interfaz gráfica, lo que implica el desarrollo de un programa con elementos visuales en lugar de uno basado exclusivamente en operaciones en consola.

En el ecosistema de desarrollo correspondiente al entorno .NET, dicha implementación generalmente se realiza mediante tecnologías como Windows Forms o WPF. Estas herramientas están específicamente diseñadas para elaborar ventanas, botones y distintos componentes visuales que conforman la interfaz del usuario.

La evidencia requerida consiste en una captura de pantalla donde sea posible apreciar de manera clara la ventana principal de la aplicación desarrollada. Dentro de esta ventana deben observarse diversos elementos gráficos, tales como botones, cuadros de texto, menús, etiquetas y cualquier otro componente que integre el diseño de la interfaz.

Asimismo, la imagen debe ilustrar algún nivel de interacción entre el usuario y la aplicación. Esto puede incluir acciones como ingresar datos en un cuadro de texto, hacer clic sobre un botón o realizar operaciones visuales que involucren algún tipo de respuesta interactiva por parte del programa.

Este requisito resulta significativo, dado que evidencia que el estudiante o desarrollador no se limitó a trabajar únicamente con aplicaciones de tipo consola, sino que adquirió capacidades adicionales para crear software con elementos visuales.

Esto no solo implica un manejo competente del entorno .NET, sino que también refleja un nivel más avanzado en las habilidades asociadas al desarrollo de aplicaciones interactivas y orientadas al usuario.

```
public partial class Form1 : Form
{
    1 referencia
    public Form1()
    {
        InitializeComponent();
    }

    1 referencia
    private void lblNombre_Click(object sender, EventArgs e)
    {
        string nombre = txtNombre.Text;
        MessageBox.Show("Oscar Esteban Sánchez Leyva");
    }

    1 referencia
    private void lblEdad_Click(object sender, EventArgs e)
    {
        string edad = txtEdad.Text;
        MessageBox.Show("31 Años");
    }

    1 referencia
    private void lblFecha_Click(object sender, EventArgs e)
    {
        string nombre = txtFecha.Text;
        MessageBox.Show("13/12/1993");
    }

    1 referencia
    private void lblCarrera_Click(object sender, EventArgs e)
    {
        string nombre = txtCarrera.Text;
        MessageBox.Show("Ingeniería en Desarrollo de Software");
    }

    1 referencia
    private void button1_Click(object sender, EventArgs e)
    {
        MessageBox.Show("Por formar parte de UMI/Universidad Coppel. Bienvenido a la carrera de Ingeniería en Desarrollo de Software.");
    }
}
```

UNIVERSIDAD MÉXICO INTERNACIONAL

Nombre Completo: **Oscar Esteban Sánchez Leyva**

Edad: **31 Años**

Fecha de Nacimiento (DD/MM/AAAA): **13/12/1993**

Carrera que desee ingresar: **Ingeniería en Desarrollo de Software**

Oscar Esteban Sánchez Leyva

Aceptar

UNIVERSIDAD MÉXICO INTERNACIONAL

Nombre Completo: **Oscar Esteban Sánchez Leyva**

Edad: **31 Años**

Fecha de Nacimiento (DD/MM/AAAA): **13/12/1993**

Carrera que desee ingresar: **Ingeniería en Desarrollo de Software**

31 Años

Aceptar

UNIVERSIDAD MÉXICO INTERNACIONAL

Nombre Completo: **Oscar Esteban Sánchez Leyva**

Edad: **31 Años**

Fecha de Nacimiento (DD/MM/AAAA): **13/12/1993**

Carrera que desee ingresar: **Ingeniería en Desarrollo de Software**

13/12/1993

Aceptar

UNIVERSIDAD MÉXICO INTERNACIONAL

Nombre Completo: **Oscar Esteban Sánchez Leyva**


Edad: **31 Años**

Fecha de Nacimiento (DD/MM/AAAA): **13/12/1993**

Carrera que desee ingresar: **Ingeniería en Desarrollo de Software**

Ingeniería en Desarrollo de Software

Aceptar

 UNIVERSIDAD MÉXICO INTERNACIONAL

Nombre Completo:

Oscar Esteban Sánchez Leyva

Edad:

31 Años

Fecha de Nacimiento (DD/MM/AAAA):

13/12/1993

Carrera que desee ingresar:

Ingeniería en Desarrollo de Software

Gracias Oscar Esteban Sánchez Leyva

X

Por formar parte de UMI/Universidad Coppel. Bienvenido a la carrera de Ingeniería en Desarrollo de Software.

Aceptar

CAPTURAS DE CADA REQUISITO SOLICITADO

El presente apartado enfatiza la importancia de proporcionar una documentación exhaustiva y precisa para la correcta entrega del proyecto, lo que implica la presentación de todas las capturas de pantalla correspondientes a los requisitos previamente especificados. Es fundamental incluir un conjunto completo y bien estructurado de evidencias visuales que reflejen el trabajo desarrollado.

Dichas capturas de pantalla deben cubrir los siguientes elementos esenciales:

1. El código en lenguaje C#, que ilustre la implementación y desarrollo del programa.
2. La consola mostrando los datos generales, evidenciando la ejecución inicial del programa.
3. La consola con los datos ingresados, donde se demuestre que el usuario proporcionó información válida y que el sistema la procesó correctamente.
4. La aplicación de escritorio, enfocándose en la interfaz gráfica creada mediante .NET, incluyendo elementos como ventanas, botones y campos de texto.

El propósito principal de este requisito es asegurar una documentación clara de cada fase del proyecto, desde la programación hasta la interacción gráfica. Esto no solo permite verificar el cumplimiento integral de las etapas del desarrollo, como la codificación, captura y procesamiento de datos, y diseño de la interfaz, sino que también promueve la transparencia y garantiza la demostración del dominio técnico sobre las herramientas empleadas.

REFERENCIAS

C#: un lenguaje de programación moderno y de código abierto / .NET. (s. f.). Microsoft.

<https://dotnet.microsoft.com/es-es/languages/csharp>

Delgado, D. O. (2017, 29 marzo). Qué es C#: Introducción. *OpenWebinars.net*.

<https://openwebinars.net/blog/que-es-c-introduccion/>

Anandmeg. (s. f.). *Creación de una aplicación de Windows Forms con C# - Visual Studio (Windows)*. Microsoft Learn. <https://learn.microsoft.com/es-es/visualstudio/ide/create-csharp-winform-visual-studio?view=visualstudio>