Administración de Tareas

Versión del Documento: 1.0

Octubre, 2025

Ing. Luis Enrique Vázquez López

Contenido

[**Versión del Documento** 3](#_Toc212669965)

[Introducción 4](#_Toc212669966)

[1. 🎯 Objetivo del Proyecto 4](#_Toc212669967)

[2. 🧭 Flujo de Uso y Menú Interactivo 4](#_Toc212669968)

[Secuencia de acciones: 4](#_Toc212669969)

[3. 🧱 Estructura del Código Fuente 6](#_Toc212669970)

[4. 💾 Persistencia de Datos 8](#_Toc212669971)

[5. 🎓 Buenas Prácticas Aplicadas 8](#_Toc212669972)

[6. 📷 Demostración del Funcionamiento 9](#_Toc212669973)

# **🔢 Versión del Documento**

|  |  |  |
| --- | --- | --- |
| **Versión** | **Modificaciones Realizadas** | **Fecha** |
| 1.0 | Draft creado por Ing. Luis Enrique Vázquez López | 29/Oct/25 |

# Introducción

Este documento contiene información de la aplicación solicitada para la administración de tareas. Aquí se detallan el objetivo, flujo de uso y estructura de código.

# 1. 🎯 Objetivo del Proyecto

El propósito principal de este proyecto es desarrollar una aplicación básica en .NET 8.0 que permita la administración de tareas desde una interfaz de consola, aplicando buenas prácticas de programación orientada a objetos (POO), estructuras de datos eficientes y un diseño claro y organizado. Esta solución está pensada como un sistema CRUD educativo y funcional, ideal para entornos de aprendizaje o como base para proyectos más complejos.

# 2. 🧭 Flujo de Uso y Menú Interactivo

Al ejecutar la aplicación, el usuario interactúa con un menú principal que ofrece las siguientes opciones:

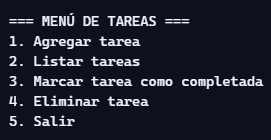


Imagen 1. Menú principal del sistema de Administración de Tareas

## Secuencia de acciones:

Opción 1: Agregar tarea

* Solicita al usuario el título de la tarea.
* Crea una nueva instancia de Tarea y la agrega a la lista.
* Muestra el mensaje: >>> Tarea agregada correctamente <<<

**Opción 2: Listar tareas**

* Muestra todas las tareas registradas con su estado ([OK] o [PENDIENTE]).
* Si no hay tareas, muestra: !!! No hay tareas registradas !!!

**Opción 3: Marcar tarea como completada**

* Muestra la lista de tareas.
* Solicita el ID de la tarea a completar.
* Cambia el estado de la tarea a completada.
* Muestra: >>> Tarea marcada como completada <<<

**Opción 4: Eliminar tarea**

* Muestra la lista de tareas.
* Solicita el ID de la tarea a eliminar.
* Elimina la tarea de la lista.
* Muestra: [- - - Tarea eliminada - - -]

**Opción 5: Salir**

* Solicita confirmación (S/N).
* Si se confirma, guarda las tareas en archivo JSON y muestra: <<< ¡Hasta pronto! >>>

# 3. 🧱 Estructura del Código Fuente

El proyecto está compuesto por tres clases principales:

**3.1 Tarea.cs**

Define el modelo de datos para cada tarea.

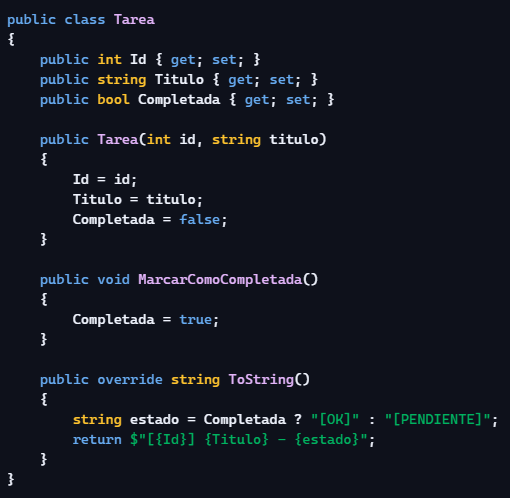


Imagen 2. Código fuente de la clase Tarea

🔹**Responsabilidad**: Representa una tarea individual con su estado.

🔹 **Métodos clave**: MarcarComoCompletada() y ToString() para visualización.

**3.2 GestorDeTareas.cs**

Administra la lista de tareas y las operaciones CRUD.

**Principales métodos:**

* AgregarTarea(string titulo): Crea y agrega una nueva tarea.
* ListarTareas(): Muestra todas las tareas.
* MarcarComoCompletada(int id): Cambia el estado de una tarea.
* EliminarTarea(int id): Elimina una tarea por ID.
* GuardarEnArchivoJson(string ruta): Serializa y guarda la lista en archivo JSON.
* CargarDesdeArchivoJson(string ruta): Carga tareas desde archivo JSON al iniciar.

🔹 **Responsabilidad**: Controlar la lógica de negocio y persistencia.

🔹 **Estructura de datos**: List<Tarea> para almacenar las tareas en memoria.

**3.3 Program.cs**

Contiene el punto de entrada y el menú interactivo.

**Flujo principal:**

* Carga tareas desde archivo JSON al iniciar.
* Muestra el menú en bucle hasta que el usuario decida salir.
* Llama a los métodos del GestorDeTareas según la opción seleccionada.
* Limpia la consola después de cada acción para mantener claridad.
* Guarda las tareas antes de salir.

🔹 **Responsabilidad**: Controlar la interacción con el usuario.

🔹 **Diseño**: Menú claro, validación de entradas, confirmación de salida.

# 4. 💾 Persistencia de Datos

El sistema guarda automáticamente las tareas en un archivo tareas.json al salir, y las carga al iniciar. Esto permite mantener el estado entre ejecuciones sin necesidad de base de datos.

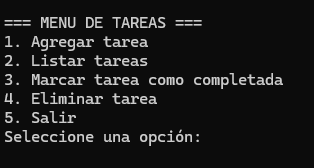
* **Formato JSON**: legible, estructurado y fácil de integrar con otros sistemas.
* **Uso de** System.Text.Json: biblioteca nativa en .NET para serialización.

# 5. 🎓 Buenas Prácticas Aplicadas

* Programación orientada a objetos: encapsulación, separación de responsabilidades.
* Código limpio y comentado.
* Validación de entradas del usuario.
* Interfaz de consola clara y profesional.
* Persistencia de datos sin base externa.
* Uso de estructuras de datos eficientes (List<T>).
* Estilo visual con delimitadores ASCII para mejorar la experiencia.

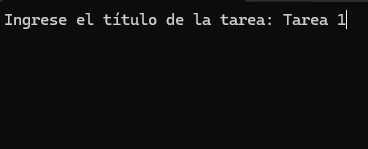
# 6. 📷 Demostración del Funcionamiento

A continuación, se describe el funcionamiento de la aplicación de Administración de Tareas.



Texto

El contenido generado por IA puede ser incorrecto.



Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.