

**Universidad Abierta y a Distancia**

**de México**

**División de Ciencias Exactas, Ingeniería y Tecnología**

**Desarrollo de software**

**Semestre:** Sexto Semestre

**Asignatura:** Programación net II

**Unidad de aprendizaje: Desarrollo de objetos en la plataforma .NET II**

**Actividad: Evidencia de aprendizaje**

**Nombre del estudiante:** Francisco González Antonio

**Matrícula:** ES1821009465

**Grupo: DS-DPRN2-2202-B2-003**

**Docente: José Francisco Rico Gallegos**

**Fecha de entrega:**

**Ciudad de México, 13 de Octubre de 2022**

Diseño: DL-CPL

**ÍNDICE**

[**INTRODUCCIÓN** 5](#_Toc80333573)

[**DESARROLLO DE LA ACTIVIDAD** 6](#_Toc80333574)

[**CONCLUSIONES** 7](#_Toc80333575)

[**REFERENCIAS BIBLIOGRÁFICAS** 8](#_Toc80333576)

# **INTRODUCCIÓN**

El caso de estudio propuesto en esta actividad permite tener un acercamiento a problemas del mundo real mediante la abstracción de objetos “reales”. Aunque esta abstracción ya estaba comentada en la actividad pues se dijeron los métodos y atributos que se requerían la implementación es el reto de esta actividad pues es necesario utilizar conceptos como la sobrecarga de métodos. La sobrecarga de métodos permite implementar diversas funcionalidades basándonos en los argumentos de entrada.

Una parte importante del desarrollo fue la creación de un pequeño menú interactivo con el usuario, de esta forma se crearon objetos de manera dinámica y con ellos ser realizaron operaciones, algo así sucedería en un entorno real a través de una aplicación móvil o una página web, ya que se pueden crear muchos objetos llamados ingredientes y todos se asocian en la elaboración de un pastel. Más aún, pensemos en que queremos tener una “tienda de ingredientes” pues sólo sería necesario tomar la clase Ingrediente e implementarla, y esto es consistente con el concepto de abstracción y el “Do not Repat Yourself” de reutilización de código.

Debido a que es importante la abstracción los métodos sobre cargados no los implementé dentro de la clase Ingrediente porque no siempre un Ingrediente se utilizará para crear un Pastel, puede crearse para un sinfín de cosas.

# **DESARROLLO DE LA ACTIVIDAD**

En esta resolución se utilizará C# con Visual Studio. Se manejarán tres clases: Clase Principal, Clase Ingrediente y Clase Pastel. Bajo el contexto de ingeniería el manejar estas clases nos permite detonar una interacción de clases donde la Clase Pastel utiliza Clases dentro de su información, aquí se vuelve relevante el concepto de NameSpace dentro de la programación ya que esto nos indica el contexto actual en el que nos encontramos y la información que estamos consumiendo.

Se espera que este programa pueda servir para crear pasteles mediante la creación de ingredientes, digamos que el alcance además de la funcionalidad es que si queremos crear Ingredientes podemos utilizar la clase independientemente de si vamos a hacer un Pastel o no.

**Requerimientos**

Los requerimientos se pueden enunciar como: crear una clase Ingrediente que es lo más molecular de este desarrollo, Crear una clase Pastel que a su vez tenga ingredientes y crear un Menú que interactúe con el usuario. Se detonan diversos requerimientos como el realizar pedido, contar calorías, determinar el costo total, mostrar el total de ingredientes y mostrar la información del pedido. Dichos métodos se estarán implementando en diferentes clases y en diferente interacción.

**Diagramas de apoyo**

Las clases solicitadas y que se implementan están a continuación:

A picture containing table

Description automatically generated

**Clase Ingrediente**

En esta clase implementé un constructor y algunos argumentos:

Text

Description automatically generated

Para ingresar la medida de rendimiento. Utilicé una validación por si algún valor se encuentra fuera de rango asigne el rendimiento mínimo:

Text

Description automatically generated

Set y get para calorías y precio:

Text

Description automatically generated

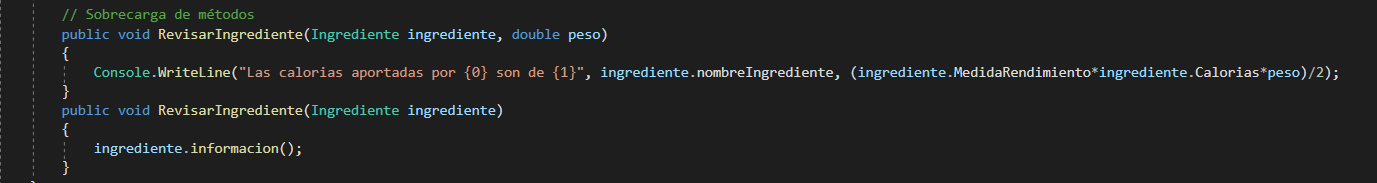
Adicional ingresé un método para regresar la información global del ingrediente:

A screenshot of a computer

Description automatically generated with medium confidence

**Método RevisarIngrediente sobre cargado**

Aquí tuve una cuestión y es que se indicaba que se sobrecargara el método, pero comprendí que el método podría estar fuera de la clase ya que no se solicita que se implemente internamente, más aún, en el caso de la clase Pastel se indica explícitamente qué métodos debe contener, por ello la programación de esta sobre carga la realicé desde la clase principal:



Y se mandan a llamar para cada ingrediente del pastel actual:

A screenshot of a computer

Description automatically generated with medium confidence

**Realizar Pedido**

Este es el método más amplio de las clases ya que aquí se utilizan los set y get de las variables como peso, rendimiento y costo. Adicional este método recibe por parámetro una lista de ingredientes del tipo Ingrediente (la otra clase desarrollada) para que este pastel ya conozca de qué va a estar hecho y en que capas.

La parte en la que se ingresan los ingredientes están en la clase principal y aunque se podían ingresar desde la clase Pastel pensé que sería mejor manejarlos aparte y que se ingresaran antes, para ello se implementó una funcionalidad dentro del menú.

Por último como se solicitó, se ingresa dentro de este método el valor de rendimiento, calorías y precio de cada ingrediente.

Text

Description automatically generated

**Calorías Totales**

Para las calorías totales tengo un método que suma las calorías de los ingredientes del pastel. No me quedó tan clara la lógica de implementación en cuanto al cómo debían interactuar los valores pero he aquí una clase implementada que nos dice las calorías totales de los ingredientes del pastel

Text

Description automatically generated

**Clase Pastel**

Adicional en la clase Pastel agregué dos métodos, el primero para determinar el costo del pastel:

Text

Description automatically generated

Text

Description automatically generated

**Cantidad de Ingredientes**

Este método permite listar los ingredientes y ver su información haciendo uso del método información de la clase Ingrediente. La funcionalidad permite al usuario saber el cómo quedó el pastel en cuando a sus ingredientes y composición, adicional permite repasar tantos ingredientes se tengan, no sólo depende de que sean 3 o 4.

Text

Description automatically generated

**Menú para interactuar**

Para el menú de interacción pensé que sería importante listar los ingredientes anexados para que el usuario sepa qué es lo que hay disponible y para que secciones del pastel:

Text

Description automatically generated

Adicional el menú muestra un flujo funcional de cómo se crea el pastel desde los ingredientes pues inicialmente se crean los ingredientes, ya que sin ingredientes no tendríamos que tener pastel:

Text

Description automatically generated

**Pantallas de prueba del programa**

Mando una prueba de la pantalla del sistema y a continuación las funcionalidades del programa:

Comenzamos con la solicitud de los ingredientes de la preparación:

Text

Description automatically generated

Por cada ingrediente se ingresa un nombre y en qué sección se colocará:

Text

Description automatically generated

Se repasan los ingredientes y se solicita la información del pastel:

Text

Description automatically generated

Después se solicitan los datos de entrada de los ingredientes, aquellos datos que se requería se ingresaran en el método realizarPedido:

Text

Description automatically generated

De nuevo enuncio la información de los ingredientes junto con los precios y las calorias:

Text

Description automatically generated

Por último respondiendo a la funcionalidad de repasar los ingredientes con la sobrecarga del método RevisarIngrediente, en la primer sección se muestran las calorías del ingrediente según las reglas proporcionadas y en la segunda se muestran las información general del ingrediente:

Text

Description automatically generated

Captura de la consola y la hora del sistema:

A picture containing text

Description automatically generated

# **CONCLUSIONES**

En esta actividad logré conocer más respecto a la sobrecarga de métodos y con ellos tener disponible diversas funcionalidades con base es los argumentos de entrada, lo que considero útil en el sentido de que se pueden tener operaciones similares pero para distintos tipos de objetos como por ejemplo calcularArea, se calcula el área de una figura geométrica pero no todos con la misma fórmula, por eso si se manda el número de lados se puede calcular pero si se pasa un radio se ocupa otra fórmula.

Considero que esta actividad es útil ya que ayuda a practicar la abstracción de objetos pues el Objeto Ingrediente se puede heredar a cualquier otra preparación sin que esta sea necesariamente un pastel, es aquí la principal razón por la que decidí poner el método sobrecargado fuera de la clase Ingrediente para que no se asociara directamente con un Pastel, pues considero que es mejor tener desasociadas las operaciones de un Ingrediente, para que el Ingrediente sea independiente al producto que se va a crear.

Creo que con esta actividad se cubren todos los temas de la unidad ya que se implementaron sobrecarga de métodos, set y get así como Constructores. No implementé el destructor porque las operaciones eran por pedido y pertinente eliminar los objetos al finalizar la ejecución, sólo ahí, no antes.

# **REFERENCIAS BIBLIOGRÁFICAS**

C. Wille; C#, Prentice Hall. 2001. ISBN 0-672-32037-1.

Deitel (2004). “Instructor Manual for Simply C#”. Pearson Education

Tanenbaum, A. (2009). “Sistemas Operativos Modernos”. 3 ed. México. Pearson Education

UnADM (2021) “Unidad 1. Introducción a .net”. Recuperado el día 27 de Septiembre de 2022 de https://campus.unadmexico.mx/contenidos/DCEIT/BLOQUE1/DS/05/DPRN1/U1/descargables/DPRN1\_U1\_Contenido.pdf