

**UNIVERSIDAD DON BOSCO**

**FACULTAD DE INGENIERÍA – ESCUELA DE COMPUTACIÓN**

**PROGRAMACION CON ESTRUCTURAS DE DATOS**

**PED104 - G01T**

**ACTIVIDAD: Primera entrega de Fase 1 de Proyecto de Catedra.**

**Fecha de entrega: 15/5/2024**

**Alumnos:**

|  |  |  |
| --- | --- | --- |
| **Nombres, Apellidos** | **Carnet** | **Fotografía** |
| Carlos Steven Martínez Navas | MN231237 |  |
| Bryan Josué Fuentes Molina | FM230331 |  |
| David Antonio Gavidia García | GG231267 |  |
| Anderson Alessandro Moya Alegría | MA230398 |  |

**ÍNDICE**

**Fase 1 de Proyecto – Contenido**

[**INTRODUCCIÓN** 3](#_Toc162039643)

[**Lógica a utilizar para el desarrollo de nuestra aplicación.** 4](#_Toc162039644)

[**Nombre del Proyecto:** Aplicación para una sucursal de venta de comida mexicana (La naca). 4](#_Toc162039645)

[**Descripción detallada de la Aplicación:** 4](#_Toc162039646)

[**Lógica a utilizar para el desarrollo de la Aplicación:** 5](#_Toc162039647)

[**Diagramas de la solución propuesta (UML).** 7](#_Toc162039648)

[**Innovación aplicada en la Aplicación y su importancia.** 9](#_Toc162039649)

[**Empleado\_Ventas:** 9](#_Toc162039650)

[**Empleado\_Almacenamiento**: 10](#_Toc162039651)

[**Gerente:** 10](#_Toc162039652)

[**Ventajas del uso de Estructuras de datos en nuestra aplicación:** 11](#_Toc162039653)

[**Nombre de nuestra Empresa / Restaurante:** 12](#_Toc162039654)

[**Prototipo de la Aplicación.** 13](#_Toc162039655)

[**Bibliografía (Fuentes de consulta)** 16](#_Toc162039656)

# **INTRODUCCIÓN**

En el contexto de un proyecto de desarrollo de software para una sucursal de venta de comida mexicana, se plantea la creación de una aplicación con dos niveles de usuario: cliente y empleado. La aplicación se diseñará para ofrecer una experiencia inmersiva y de calidad, mejorando así el impacto y la eficiencia de la empresa en cuestión.

Para lograr estos objetivos, se emplearán tecnologías de vanguardia como el .NET Framework más actualizado, así como el uso de herramientas como Windows Forms para el diseño de la interfaz de usuario. Además, se utilizará SQL Server Management Studio para el manejo de la base de datos relacional, aprovechando las ventajas que ofrece en cuanto a gestión de existencias, control de productos y seguridad de datos.

El sistema contará con distintos roles, como el de gerente, empleado de ventas, empleado de almacenamiento y cliente, cada uno con sus propias autenticaciones y privilegios específicos. Esto permitirá una gestión eficiente de las ventas, el control de las existencias, la administración de promociones y la mejora continua de los procesos internos.

Uno de los aspectos destacados del proyecto es el uso de estructuras de datos (Listas, colas, etc) para el manejo de datos, como listas para el carrito de compras de los clientes, colas para la gestión de ventas y almacenamiento, y la implementación de algoritmos eficientes para garantizar la calidad y disponibilidad de los productos.

Pero no solamente se contará con eso, además se contará con seguridad de encriptación Hash, para que se puedan acceder a Menús que sean únicos en la aplicación, así como también promociones que son de determinadas fechas, se necesitara también posiblemente el uso de un ABB, para la visualización de las existencias próximas a vencer.

# **Lógica a utilizar para el desarrollo de nuestra aplicación.**

## **Nombre del Proyecto:** Aplicación para una sucursal de venta de comida mexicana (La naca).

## **Descripción detallada de la Aplicación:**

La aplicación contara con dos niveles de usuario, uno para clientes y otro para empleados. Se ha decidido trabajar de esta manera ya que se tiene más libertad a la hora de utilizar estructuras de datos en la aplicación, también por la necesidad que tiene el proyecto de mejorar la calidad de trabajo de los empleados que tienen un rol en la aplicación. Por ejemplo, se pretende crear un carrito de compras para la aplicación, este mismo funcionaria básicamente para la realización de las compras dentro del sistema, en este carrito se utilizaría la estructura de datos **Lista**, para que el cliente tenga la libertad de agregar, modificar o eliminar ya sea a su gusto. Prosiguiendo, para el nivel de usuario del empleado, habrá dos apartados: **uno de ventas y otro de almacenamiento**. En ventas se ocupará la estructura de datos llamada **Cola**, por la prioridad que tiene de **LIFO**. En almacenamiento, también se empleará el uso de un ABB, ya que en la aplicación manejaremos existencias, es decir, que habría productos en **Stock** y dependiendo de esos productos y el **tipo que sean**, se realizarían los recuentos de las existencias por **fecha de vencimiento.** Otra de las razones por las cuales utilizaremos ABB (ya profundizando, tipo **heap minimizante** por su característica de mandar valores mayores al fondo, y que permanezcan valores menores arriba) para el apartado de almacenamiento es para cuando las existencias estén por caducar, ya que se tendrá que codificar que ABB posea una fecha determinada y que esta no limite después de dicha fecha, para la no comercialización de productos en mal estado o caducados, todo esto gracias en parte al tipo que es ese producto y la especialidad.

Luego de explicado esto entra el nivel de usuario que puede entrar en el de empleado, que es el **Gerente,** el Gerente tendrá un acceso completo a todo lo que es la aplicación, esto implica que los empleados anteriores junto con el cliente no tendrán un acceso totalitario en la aplicación como es evidente, ahora bien, la vista del Gerente también tendrá su propia manipulación dentro de la aplicación, como se menciono antes, podrá acceder al Menú, donde podrá modificar los precios, pero esto tiene que tener un nivel de seguridad como es lógico, por que constara de una encriptación Hash, se explica más detalladamente en el otro inciso.

La aplicación en general, constara de roles, como el de **Gerente**, el **Empleado de ventas**, el **Empleado de almacenamiento**, y el mas importante, el **Cliente**. Todos ellos contarían con su propia autenticación en la aplicación para mejorar la seguridad de este mismo.

## **Lógica a utilizar para el desarrollo de la Aplicación:**

Se utilizará el **Net.Framework** más actualizado, que puede no ser de mucha importancia, pero será de ayuda para la optimización y mejora en el rendimiento de la aplicación, recomendándole al usuario, por consiguiente, que también lo tenga actualizado. Volviendo al tema de estructuras de datos, se pretende utilizar también **Tablas Hash**.

Para el manejo del Sistema de base de datos, se ocupará como gestor: **SQL server Management Studio**, debido a la facilidad que tenemos de manejar este programa. Se ocupará una base de datos relacional, donde existirán relaciones de uno a muchos (**claves foráneas, claves primarias**); esto nos será de gran ayuda a la hora de manejar las existencias de la aplicación y también para controlar las propiedades de los productos ingresados a la aplicación.

También porque será nuestro pilar a la hora de programar, porque de ahí derivaremos nuestros algoritmos. Para ello, nos será de utilidad un **CRUD**, para llevar a cabo el **Entity Framework.** Se ocupará el Entity Framework, por su rapidez y simplificación del CRUD; para evitarse así, el tener código extenso en la aplicación y también para ahorrarnos tiempo de programación.

El diseño de nuestra aplicación, contando con los niveles de usuario, contendrá también con las herramientas básicas que proporciona **Windows Forms,** no se utilizaría ningún agente externo, u otro tipo de lenguaje de ensamblado, ya que no vemos necesaria la intervención de los mismos; más, sin embargo, esto no implica que quizás en el futuro no se tenga la idea de implementarse alguno de estos, en otras palabras, nuestro proyecto se guiara primeramente en todo lo que ofrece Windows Forms.

Se pretende utilizar la mítica por todo Ingeniero en Sistemas **prueba y error,** con el fin de proporcionar una aplicación funcional en todos los rasgos ya descritos anteriormente.

La lógica enteramente ira en nuestras estructuras de datos, con protagonismo en los nodos de las mismas, se plantea la creación de varios nodos para distintas colas y listas, porque cada una tendrá una propiedad que la diferencie de la otra, como la lista del carrito que no necesita de un nodo que le calcule si la fecha del producto está por expirar, entre otras cosas.

# **Diagramas de la solución propuesta (UML).**

Diagrama de Casos de Uso:

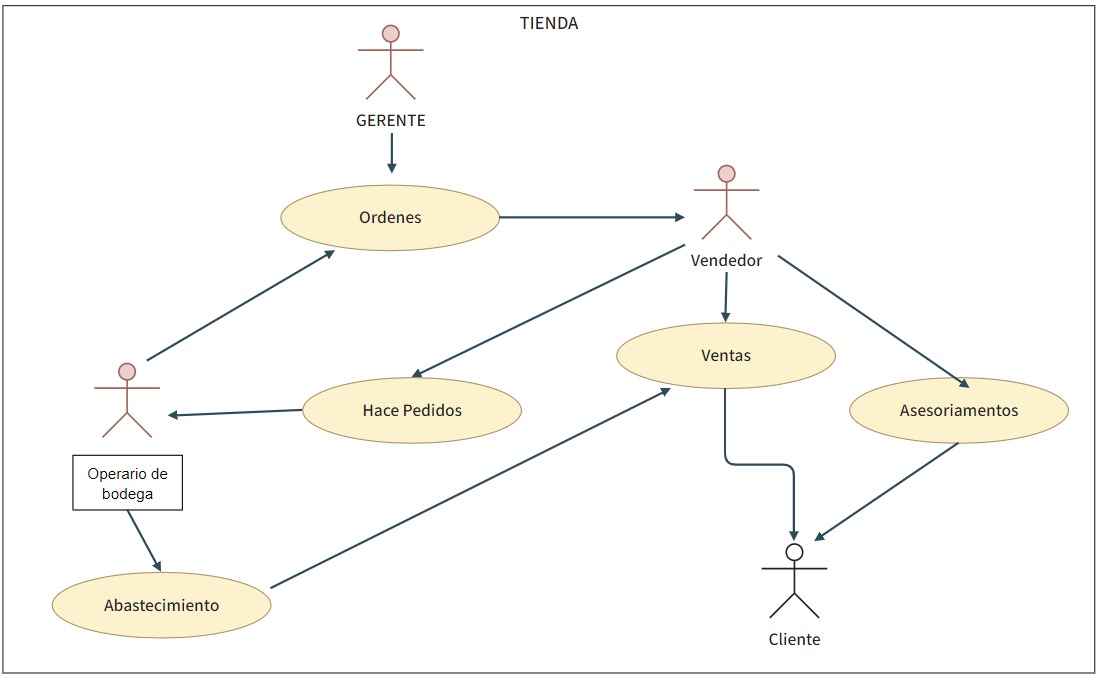


Diagrama de Clases

Diagrama

Descripción generada automáticamenteDiagrama

Descripción generada automáticamente

# **Innovación aplicada en la Aplicación y su importancia.**

La innovación viene radicada en las técnicas de programación que tendrá nuestra aplicación, así como su idea de mejorar el impacto en calidad de nuestra empresa. Los propósitos del proyecto se resumen en: **inmersividad, disponibilidad y calidad.** Para ello, como se mencionó en anteriores incisos, se ocuparan estructuras de datos para el manejo de los mismos, con propiedades especiales para cada uno de sus propios nodos, para ello se mira necesario la creación desde cero de cada tipo de estructura, para tener control total de la misma y para así facilitar el trabajo de los usuarios empleados, ¿cómo se facilitara y como se beneficiara exactamente el empleado?, por que haremos fácil tres caminos, que estos tres caminos vienen de la mano con los roles existentes en la aplicación:

## **Empleado\_Ventas:**

Gozara de acceso completo a todo el sistema de ventas entrantes en la aplicación, en tiempo real emergentes en el restaurante La Naca, esto gracias a la estructura cola**\_ventas**, que mediante el control de formulario: **Label** ira mostrando gráficamente cada venta entrante en orden de prioridad. Con orden de prioridad se refiera a que se dará prioridad a pedidos de categoría "simple". En ventas de categoría "simple" entraran los clientes que solamente piden un tipo de producto fácil, junto con una bebida fácil. La cantidad también es determinante al evaluar la categoría "simple" en ventas, si se supera la cantidad de 6 unidades en nuestros productos de tipo fácil, se cambiará su orden de categoría a una categoría "media". En la Categoría tipo "media": entran aquí las ventas que contenga 6 unidades o más de producto tipo fácil, o que contenga 1 producto de tipo medio junto con 1 o 2 productos de tipo fácil. La Categoría de tipo "Alto" seria: cualquier tipo de producto que se exceda de 5 o 6 unidades, también si se tiene 3 productos de tipo medio. Las fechas de entrega serán determinadas acordes a: La cantidad de producto pedido y la categoría de tipo de venta, se le mostrará al cliente, una vez el encargado de ventas acepte el tipo de entrega, donde el mismo le dará fecha a la venta de entrega. (Colores para categorías irían: verde para simple, amarillo para medio y rojo para alto.)

## **Empleado\_Almacenamiento**:

Tendrá un acceso completo a los productos existentes en la aplicación, junto con todas las características importantes del producto, gozará de que la aplicacion le ordenará las existencias acordes fecha de caducidad.

Tipo de existencias:

**Facil:** Facil, son los productos que: o no son cocinados en nuestra instalación o son de preparamiento rápido con ejemplos claros de: las bebidas enlatadas, tacos o baleadas.

**Medio:** Son los productos que se requiere una mayor cantidad de ingredientes, mayor tiempo de procesamiento, ej.: Tortas, tortas ahogadas.

**Alto:** Son pedidos que son familiares, o que son de carácter festivo o de alguna promoción.

El empleado también contará con un apartado para ver las fechas de caducidad de los productos, con fecha de caducidad se refiere a la materia con la que se crea el producto, por ejemplo, la carne, la carne es necesaria para hacer cualquier taco, pues esa carne tendrá una fecha límite donde pueda estar en nuestras instalaciones, superada esa fecha, en la aplicación se mostrara que ha caducado. También podrá ingresar directamente a la base de datos existencias a la aplicación, con el fin de mejorar la productividad del sistema.

## **Gerente:**

Gozará del acceso completo a toda la aplicación, se constará de una encriptación Hash, donde el gerente sea el único que tenga el acceso a las Listas de clave, donde él pueda ingresar la lista de clave y ver el Menú especifico, y así poder cambiar su precio, su nombre etc. Podrá crear productos y esos productos creados se guardaran en la base de datos

## **Ventajas del uso de Estructuras de datos en nuestra aplicación:**

**Eficiencia y rendimiento mejorados:** Las estructuras de datos eficientes son fundamentales para garantizar un rendimiento óptimo de las aplicaciones. La innovación en este campo implica el desarrollo de algoritmos más rápidos y optimizados, así como la creación de nuevas estructuras de datos que permitan una manipulación y acceso más eficientes a los datos. Esto es especialmente importante en aplicaciones que manejan grandes volúmenes de datos o que requieren tiempos de respuesta rápidos.

**Escalabilidad**: Con el crecimiento constante de la cantidad de datos que se manejan en las aplicaciones modernas, es crucial que las estructuras de datos puedan escalar de manera efectiva. La innovación en este ámbito implica el diseño de estructuras de datos que puedan manejar grandes volúmenes de datos sin sacrificar el rendimiento o la eficiencia.

**Facilita la resolución de problemas complejos**: Muchos problemas en informática requieren el uso de estructuras de datos sofisticadas para su resolución. La innovación en este campo puede llevar al desarrollo de nuevas estructuras de datos o algoritmos que simplifiquen la resolución de estos problemas, permitiendo a los programadores abordar desafíos cada vez más complejos de manera más efectiva.

**Optimización de recursos:** La innovación en programación con estructuras de datos también puede contribuir a la optimización de recursos como memoria, procesamiento y ancho de banda. Esto es especialmente importante en entornos con recursos limitados, como dispositivos móviles o sistemas embebidos, donde el uso eficiente de los recursos es esencial.

**Mejora la calidad del software:** Utilizar estructuras de datos adecuadas y técnicas innovadoras en su implementación puede mejorar la calidad general del software. Esto incluye aspectos como la robustez, la mantenibilidad, la extensibilidad y la facilidad de depuración del código.

**Fomenta la creatividad y la exploración:** La innovación en programación con estructuras de datos fomenta la creatividad y la exploración en el campo de la informática. Permite a los desarrolladores experimentar con nuevas ideas y enfoques, lo que puede conducir a descubrimientos importantes y avances significativos en la forma en que se gestionan y manipulan los datos.

# **Nombre de nuestra Empresa / Restaurante:**

Nuestra Empresa / Restaurante tendrá como nombre “La Naca”. Restaurante de comida mexicana (ventas y almacenamiento).



# **Prototipo de la Aplicación.**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Escala de tiempo

Descripción generada automáticamente**

****

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

# **Bibliografía (Fuentes de consulta)**

* <https://ed.team/blog/estructuras-de-datos>
* <https://learn.microsoft.com/es-es/troubleshoot/developer/visualstudio/csharp/language-compilers/hashtable-collection>