

Aplicacion de gestion de pedidos para el restaurante Casa Dorada

Presentado por:

Juan David Cruz Garcia

Samuel Hernandez Cifuentes

Algoritmos y Programación II - Ingeniería de Sistemas

Universidad Icesi

Cali, Valle del Cauca

Primer periodo academico

2021

## Requerimientos Funcionales

El sistema debe estar en la capacidad de:

**Req 1. Gestionar** los productos del restaurante. Los productos cuentan con un nombre, un tipo de producto, una lista de ingredientes y una lista de tamaños con sus respectivos precios.

- Req 1.1 Agregar productos al restaurante. El tipo de producto y los ingredientes del producto a crear deben pertenecer a aquellos que ya se encuentran en el restaurante y deben estar habilitados. No se puede agregar más de un producto con el mismo nombre.
- Req 1.2 Eliminar productos del restaurante. Un producto podrá ser eliminado si y solo si no está referenciado desde un pedido.
- Req 1.3 Deshabilitar un producto. Esto implica que el producto no podrá ser usado desde el momento en que se deshabilita en adelante.
- Req 1.4 Habilitar un producto. Lo cual permite que un producto deshabilitado pueda volver a ser usado.
- Req 1.5 Actualizar los datos del producto.
  - Req 1.5.1 Cambiar el nombre del producto. No debe ser posible cambiar el nombre del producto actual para que coincida con el nombre de otro producto en el restaurante.
  - Req 1.5.2 Cambiar el tipo de producto del producto. Solo será posible si el nuevo tipo de producto está habilitado en el restaurante.
  - Req 1.5.3 Agregar un ingrediente al producto. El ingrediente que se agrega debe existir en el restaurante y debe estar habilitado.
  - Req 1.5.4 Borrar un ingrediente del producto. Esto será posible mientras no sea el último ingrediente que tiene el producto.

- Req 1.5.5 Cambiar el nombre de los tamaños del producto. No se puede cambiar el nombre a uno que ya exista entre los tamaños del producto
- Req 1.5.6 Cambiar el precio de los tamaños del producto

**Req 2. Gestionar** los ingredientes del restaurante. Los ingredientes cuentan únicamente con un nombre.

- Req 2.1 Agregar un ingrediente al restaurante. El nombre del ingrediente que se agrega no puede estar repetido.
- Req 2.2 Eliminar ingredientes del restaurante. Esto solo será posible si estos no están referenciados desde un producto.
- Req 2.3 Deshabilitar un ingrediente. Esto implica que el ingrediente no podrá ser usado desde el momento en que se deshabilita en adelante
- Req 2.4 Habilitar un ingrediente. Lo cual permite que un ingrediente deshabilitado pueda volver a ser usado.
- Req 2.5 Cambiar el nombre del ingrediente. No debe ser posible cambiar el nombre del ingrediente actual para que coincida con el nombre de otro ingrediente en el restaurante.

**Req 3. Gestionar** los tipos de plato del restaurante. Los tipos de plato cuentan únicamente con un nombre.

- Req 3.1 Agregar un tipo de producto al restaurante. El nombre del tipo de producto que se agrega no puede estar repetido.
- Req 3.2 Eliminar tipos de plato del restaurante. Esto solo será posible si estos no están referenciados desde un producto.
- Req 3.3 Deshabilitar un tipo de producto. Esto implica que el tipo de producto no podrá ser usado desde el momento en que se deshabilita en adelante

- Req 3.4 Habilitar un tipo de producto. Lo cual permite que un tipo de producto deshabilitado pueda volver a ser usado.
- Req 3.5 Cambiar el nombre del tipo de producto. No debe ser posible cambiar el nombre del tipo de producto actual para que coincida con el nombre de otro tipo de producto en el restaurante.

**Req 4. Gestionar** a los clientes del restaurante. Los clientes cuentan con un nombre, un apellido, un número de identificación (sólo si los clientes deciden proporcionarlo), una dirección, un número de teléfono y un campo de observaciones que otorga información acerca de este.

- Req 4.1 Agregar un cliente al restaurante. Dado a que el número de identificación es opcional, se espera que en la información que se agrega, el nombre y el apellido del cliente no coincidan exactamente con el nombre y apellido de otro cliente en el restaurante.
- Req 4.2 Eliminar un cliente del restaurante. Esto solo será posible si el cliente no se encuentra referenciado por un pedido.
- Req 4.3 Deshabilitar un cliente. Esto implica que el cliente no podrá ser usado desde el momento en que se deshabilita en adelante
- Req 4.4 Habilitar un cliente. Lo cual permite que un cliente deshabilitado pueda volver a ser usado.
- Req 4.5 Actualizar los datos del cliente
  - Req 4.5.1 Cambiar el nombre del cliente. Si el nuevo nombre para este cliente coincide con el nombre de otro cliente y además, comparten el mismo apellido, no se podrá realizar esta acción. Ejemplo: El nombre inicial del cliente es Pablo Castillo, si en el restaurante existe un cliente llamado Juan Castillo, no se podrá cambiar el nombre de Pablo a Juan

- Req 4.5.2 Cambiar el apellido del cliente. Si el nuevo apellido para este cliente coincide con el apellido de otro cliente y además, comparten el mismo nombre, no se podrá realizar esta acción. Ejemplo: El nombre inicial del cliente es Pablo Castillo, si en el restaurante existe un cliente llamado Pablo Carrero, no se podrá cambiar el apellido de Castillo a Carrero.
- Req 4.5.3 Cambiar el número de identificación del cliente. En caso de que inicialmente el cliente no hubiera proporcionado su identificación, esta opción servirá para registrar este número por primera vez.
- Req 4.5.4 Cambiar la dirección del cliente.
- Req 4.5.5 Cambiar el numero de telefono del cliente
- Req 4.5.6 Cambiar las observaciones del cliente

**Req 5. Gestionar** a los empleados del restaurante. Los empleados cuentan con un nombre, un apellido y un número de identificación.

- Req 5.1 Agregar empleados al restaurante. No se pueden agregar empleados con el mismo numero de identificacion.
- Req 5.2 Eliminar empleados del restaurante. Esto solo será posible si el empleado no está referenciado desde algún pedido.
- Req 5.3 Deshabilitar un empleado. Esto implica que el empleado no podrá ser usado desde el momento en que se deshabilita en adelante
- Req 5.4 Habilitar un empleado. Lo cual permite que un empleado deshabilitado pueda volver a ser usado.
- Req 5.5 Actualizar los datos de un empleado
  - Req 5.5.1 Cambiar el nombre de un empleado
  - Req 5.5.2 Cambiar el apellido de un empleado

- Req 5.5.3 Cambiar el número de identificación del empleado. No debe ser posible cambiar el número de identificación actual para que coincida con el número de identificación de otro empleado en el restaurante.

**Req 6. Gestionar** los pedidos del restaurante. Los pedidos cuentan con un código autogenerado, una lista de productos del restaurante, las respectivas cantidades especificadas para cada producto pedido, el cliente que solicitó el pedido, el empleado que realizará la entrega del pedido y algunas observaciones a tener en cuenta.

- Req 6.1 Crear un pedido. Tanto los productos como el cliente y el empleado del pedido deben existir en el restaurante y deben estar habilitados.
- Req 6.2 Cambiar el estado del pedido. Existen 5 tipos de estado SOLICITADO, EN PROCESO, ENVIADO y ENTREGADO, o también CANCELADO. El estado del pedido puede ser cambiado a estados posteriores uno a uno pero no puede ser cambiado a estados inferiores (Ejemplo: Un pedido puede pasar de ENVIADO A ENTREGADO pero no de ENVIADO a EN PROCESO)

**Req 7. Gestionar** a los usuarios del restaurante. Los usuarios se crean en base a un empleado existente en el restaurante y tienen como atributos propios un nombre de usuario y una contraseña.

- Req 7.1 Crear el usuario. Al momento en que se crea un empleado, el usuario tendrá como nombre de usuario predeterminado el nombre del empleado y como contraseña el número de identificación del empleado. El programa tiene la opción de inicializar el usuario indicando el nombre de usuario escogido y la contraseña que se desea utilizar.
- Req 7.2 Cambiar el nombre de usuario. No debe ser posible cambiar el nombre de usuario actual para que coincida con el nombre de usuario de otro usuario en el restaurante.
- Req 7.3 Cambiar la contraseña de un usuario.

- Req 7.4 Deshabilitar un usuario. Esto implica que el usuario no podrá ser usado desde el momento en que se deshabilita en adelante
- Req 7.5 Habilitar un empleado. Lo cual permite que un usuario deshabilitado pueda volver a ser usado.

**Req 8. Permitir** a los usuarios ejecutar las acciones de la aplicación.

- Req 8.1. Permitir a los usuarios iniciar sesión en la aplicación.
- Req 8.2 Mostrar en cada objeto creado, el nombre de usuario de aquel que lo creo.
- Req 8.3 Mostrar en cada objeto modificado, el nombre de usuario de aquel que lo modificó.
- Req 8.4 Permitir a los usuarios hacer uso de cada una de las funcionalidades ya especificadas y las que se especificarán a lo largo de este documento.

**Req 9. Buscar** eficientemente un cliente dado un nombre y un apellido. El programa debe devolver el tiempo en nanosegundos que tardó la búsqueda.

**Req 10. Implementar** al menos dos métodos de ordenamiento de los vistos en clase.

(Burbuja, inserción y selección)

**Req 11. Implementar** al menos un método de ordenamiento usando Comparable y un método de ordenamiento usando Comparator. En ambos casos, haciendo uso del Sort de Collections o de Arrays.

**Req 12. Generar** reportes específicos

- Req 12.1. Generar un archivo csv de pedidos, con una fila por cada pedido, con los datos del nombre, dirección y teléfono del cliente que lo solicitó, el nombre del empleado que lo entrega, estado del pedido y además de los demás datos del pedido como la fecha y hora, y las observaciones, debe tener tres columnas por cada producto del pedido con el nombre, la cantidad y el valor unitario del producto

- Req 12.1.1 El reporte debe ser generado en un rango de fechas y hora, es decir, al momento de generarlo se pregunta la fecha y hora inicial, y la fecha y hora final del reporte
- Req 12.1.1 El reporte debe usar como separador aquel que el usuario indique.
- Req 12.2 Generar un listado de empleados consolidando el número de pedidos entregados y la suma de los valores de dichos pedidos
  - Req 12.2.1 El reporte debe ser generado en un rango de fechas y hora, es decir, al momento de generarlo se pregunta la fecha y hora inicial, y la fecha y hora final del reporte
- Req 12.3 Generar un listado de los productos consolidando el número de veces que se pidió y la cantidad de dinero total que se pagó por todos los productos.
  - Req 12.3.1 El reporte debe ser generado en un rango de fechas y hora, es decir, al momento de generarlo se pregunta la fecha y hora inicial, y la fecha y hora final del reporte

**Req 13. Listar** en pantalla todos los productos en orden de precio ascendente.

**Req 14. Listar** en pantalla todos los ingredientes en orden alfabético descendente.

**Req 15. Importar** datos de distintos archivos csv de prueba generados por aplicaciones externas.

- Req 15.1 Importar datos desde un archivo con información de clientes
- Req 15.2 Importar datos desde un archivo con información de productos
- Req 15.2 Importar datos desde un archivo con información de pedidos. Para que esto sea posible se deben importar los datos de los archivos de clientes y productos.

**Req 16. Guardar** toda la información del programa a través de la serialización de los objetos de este en archivos. Estos deben ser transparentes para el usuario del programa, es decir, cada que se registre o actualice información, esta se verá reflejada en los archivos serializados.



**Req 17. Generar** un formulario para cualquier objeto (productos, pedidos, clientes, usuarios, etc). Estos formularios deben contener la información más relevante del producto.

- Req 17.1 Modificar un registro si se hace doble clic en la fila en la que se encuentra

**Req 18. Integrar** las imágenes y colores del restaurante.

## **Diagrama de clases**

