

## Requerimientos funcionales: Tarea Integradora 1

El programa debe estar en capacidad de:

**RF1: Crear** un objeto (producto, ingrediente, tipo de producto).

**RF2: Actualizar** la información que provee los objetos (producto, ingrediente, tipo de producto).

**RF3: Eliminar** un objeto (producto, ingrediente, tipo de producto) siempre que no esté referenciado desde otro objeto. Por ejemplo, un ingrediente podrá ser eliminado siempre que no haya un producto que esté asociado a dicho ingrediente.

**RF4: Deshabilitar** un objeto siempre será posible y esto implicará que ese producto no estará disponible, para ser referenciado desde otro objeto. Por ejemplo, si se deshabilita un producto, este ya no se mostrará entre los productos disponibles al momento de hacer un pedido.

**RF5: Gestionar productos** en donde todo producto tiene un nombre, un tipo (plato principal, adicional o bebida ), un conjunto de ingredientes, y tamaños con sus respectivos precios.

**RF5.1: Crear** un producto que ofrece el restaurante.

**RF5.2: Actualizar** un producto de los que ofrece el restaurante, por si hay algún cambio.

**RF5.4: Deshabilitar** un producto por si no está disponible.

**RF5.3: Eliminar** un producto por si ya no se seguirá ofreciendo.

**RF6: Gestionar clientes** ellos deben tener nombres, apellidos, un número de identificación (opcional), dirección, teléfono y un campo de observaciones. El número de identificación no puede ser repetido en ningún otro cliente, el programa no dejará registrar ninguna persona si este es el caso.

**RF6.1: Crear** un cliente.

**RF6.2: Actualizar** la información del cliente.

**RF7: Gestionar empleados** ellos al igual que los clientes deben tener nombres, apellidos, un número de identificación. El número de identificación no puede ser repetido en ningún otro cliente, el programa no dejará registrar ninguna persona si este es el caso.

**RF7.1: Crear** un empleado por si fue contratado para trabajar en el restaurante.

**RF7.2: Actualizar** la información de un empleado.

**RF7.3: Deshabilitar** un empleado por si no se encuentra trabajando actualmente.

**RF7.4: Eliminar** un empleado por si fue despedido.

**RF8: Gestionar usuarios** todo usuario es empleado, por lo tanto, también tienen nombres, apellidos, un número de identificación, un nombre de usuario y una contraseña para ingresar al sistema. El número de identificación y el nombre de usuario no pueden ser repetidos en ningún usuario, el programa no dejará registrar ninguna persona si este es el caso.

**RF8.1: Crear** un nuevo usuario con sus datos personales (nombre, apellidos y el ID) además con un nombre de usuario único y una contraseña.

**RF8.2: Actualizar** la información del usuario, por si cambió su nombre, su apellido (el número de identificación no se puede actualizar) su nombre de usuario o su contraseña.

**RF8.3: Eliminar** un usuario por si ha dejado de pertenecer al restaurante.

**RF8.4: Inhabilitar** un usuario mientras esté inactivo del restaurante.

**RF9: Ordenar** la lista de clientes alfabéticamente descendente por apellido y nombre, por lo tanto, cada vez que se agrega un nuevo cliente, este debe insertarse de forma ordenada.

**RF10: Registrar pedidos** que tienen un código, un estado, un listado de productos, la cantidad por cada uno, el cliente que los solicita, el empleado que lo entrega, la fecha y hora de la solicitud, y las observaciones que pueda tener. El programa debe tomar la fecha y hora de la solicitud, de la fecha y hora actual del sistema

**RF11: Cambiar el estado de un producto** entre SOLICITADO, EN PROCESO, ENVIADO y ENTREGADO. Es importante tener en cuenta, que solo se puede ir hacia adelante, es decir, solo se puede pasar de “solicitado” a “en proceso” y no hacia atrás.

**RF12: Listar** cualquier objeto (productos, pedidos, clientes, usuarios, etc.), donde el programa debe tener la posibilidad, de visualizar un listado de ellos, con columnas mostrando sus principales campos

**RF13: Modificar un registro** si se hace doble clic en la fila en la que se encuentra. Todo objeto, de las clases del modelo, debe tener dos campos internos referenciados a un objeto usuario, uno al usuario que lo creó y otro al último usuario que lo modificó.

**RF14: Guardar información** a través de la serialización de sus objetos en archivos. Este guardado debe ser transparente para el usuario del programa, es decir, cada vez que se registre o actualice información, esta se guardará en los archivos serializados

**RF15: 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 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. Las columnas de los productos van al final porque cada pedido tiene una cantidad diferente de productos

**RF16: Ordenar archivo csv**, debe poder generarse 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. Por defecto, el valor en la fecha y hora inicial son las 00:00 del día actual, y la fecha final debe ser por defecto 23:59 del día actual. El reporte debe estar ordenado por fecha y hora del pedido ascendente

**RF17: Exportar información** que irá al archivo csv generado, para eso también se le preguntará al usuario cuál es el separador que se utilizará, aunque por defecto debe tener puesto en ese campo el valor punto y coma (;) . La primera línea del archivo debe tener los nombres de las columnas separadas también por dicho separador, menos las columnas de los productos.

**RF18: Generar listado de empleados** consolidando el número de pedidos entregados y la suma de los valores de dichos pedidos. Esto en un rango de fecha y hora, inicial y final. Por defecto, el valor en la fecha y hora inicial son las 00:00 del día actual, y la fecha final debe ser por defecto 23:59 del día actual. Al final se debe agregar una fila que totalice las columnas numéricas.

**RF19: Generar listado de productos** consolidando el número de veces que se pidió y la cantidad de dinero total que se pagó por todos los productos (número de veces por precio del producto). Este reporte también debe solicitar un rango de fechas inicial y final igual que el anterior, con los mismos valores por defecto. Estas fechas y horas siempre se podrán cambiar. Al final se debe agregar una fila que totalice las columnas numéricas.

**RF20: Listar** en pantalla todos los productos en orden de precio ascendente

**RF21: Listar** en pantalla todos los ingredientes en orden alfabético descendente.

**RF22: Buscar eficientemente** un cliente dado un nombre e indicar el tiempo que tardó la búsqueda. Esta opción debe estar en el campo que permite realizar los pedidos.

**RF23: Importar** datos de un archivo csv con información de clientes.

**RF24: Importar** datos de un archivo csv con información de productos.

**RF25: Importar** datos de un archivo csv con información de pedidos.

**RF26: Generar** un archivo csv de prueba con al menos 1000 datos para probar los productos

**RF27: Generar** un archivo csv de prueba con al menos 1000 datos para probar los pedidos

**RF28: Generar** un archivo csv de prueba para los clientes.