

Programación II

Evaluación 1 (20% del ramo)

Problema

Estamos desarrollando una aplicación para un restaurante que permita gestionar **ingredientes, stock, pedidos y generar boletas**. La solución debe proporcionar una interfaz gráfica para los usuarios, permitir la gestión de pedidos con diferentes productos y generar boletas detalladas en formato PDF que puedan ser impresas o guardadas.

Se ha dispuesto un video <https://www.youtube.com/watch?v=VO07irdyyvs> con el software en funcionamiento para que puedan revisar con más detalle.

los menús para ofrecer están fijados por el cliente y son:

Menús	precio de venta	Ingredientes
Papas fritas	500	5 x papas
Pepsi	1100	1 x bebida
Completo	1800	1x vienesa 1x pan de completo 1x tomate 1x palta
Hamburguesa	3500	1x pan de hamburguesa 1x lamina de queso 1x churrasco de carne

Ingreso de Ingredientes

Propósito: La pestaña 1, titulada "Ingreso de Ingredientes" (ver video), está diseñada para permitir a los usuarios gestionar el stock de ingredientes. A través de esta interfaz, los usuarios pueden añadir nuevos ingredientes, eliminar ingredientes existentes, y visualizar la lista actual de ingredientes disponibles. (si tiene dudas ver el video con programa funcionando <https://www.youtube.com/watch?v=VO07irdyyvs>)

Componentes:

1. Formulario de Ingreso de Ingredientes:

- **Campo "Nombre del Ingrediente":**
 - Un campo de entrada (CtkEntry) donde el usuario puede escribir el nombre del ingrediente que desea agregar al stock.
- **Campo "Cantidad":**

- Un segundo campo de entrada (CTkEntry) donde el usuario especifica la cantidad del ingrediente que quiere agregar.
- **Botón "Ingresar Ingrediente":**
 - Un botón (CTkButton) que, al ser presionado, valida los datos ingresados y añade el ingrediente al stock. Si el ingrediente ya existe, se actualiza su cantidad. (Recuerda que como estamos bajo el paradigma de orientación a objetos debes instanciar un objeto de tipo ingredientes y agregarle la información obtenida en el formulario)
- 2. **Lista de Ingredientes:**
 - **Treeview:**
 - Un componente Treeview que muestra una tabla con dos columnas: "Nombre" y "Cantidad". Aquí se listan todos los ingredientes que están actualmente en el stock, permitiendo a los usuarios ver de un vistazo los ingredientes disponibles y sus cantidades.
 - **Botón "Eliminar Ingrediente":**
 - Este botón (CTkButton), situado justo encima del Treeview y alineado a la derecha, permite al usuario seleccionar un ingrediente de la lista y eliminarlo del stock. (recuerda que al eliminar el ingrediente de la interfaz también debes eliminarlo del objeto)
- 3. **Botón "Generar Menú":**
 - **Ubicación:**
 - Este botón se encuentra debajo del Treeview.
 - **Función:**
 - Al presionarlo, se genera un conjunto de menús basados en los ingredientes disponibles en el stock en la **pestaña Pedido** (ver video). Si faltan ingredientes para algún menú, se notifica al usuario con un mensaje de advertencia.

Validaciones:

- **Nombre del Ingrediente:**
 - El nombre debe contener solo letras y espacios. Se realiza una validación para asegurar que no se ingresen números o caracteres especiales (ver video)
- **Cantidad:**
 - Debe ser un número entero positivo. Se verifica que el usuario ingrese un valor válido antes de procesar la entrada.

Interacción con el Usuario:

- Cuando el usuario ingresa un ingrediente y presiona "Ingresar Ingrediente", el sistema valida la entrada, actualiza el Treeview para reflejar el nuevo ingrediente o la cantidad actualizada, y actualiza el stock.
- Al eliminar un ingrediente mediante el botón "Eliminar Ingrediente", el ingrediente seleccionado es removido del stock y del Treeview.

- El botón "Generar Menú" revisa los ingredientes en el stock y crea menús disponibles en la pestaña2, **notificando al usuario si algún menú no puede ser preparado debido a la falta de ingredientes.**

Descripción de la Pestaña 2: Pedido

Propósito: La pestaña 2, titulada "**Pedido**", está diseñada para gestionar la selección de menús a partir de los ingredientes disponibles en el stock. Esta pestaña permite a los usuarios ver qué menús pueden prepararse, añadirlos a un pedido, gestionar las cantidades, y finalmente, generar una boleta en formato PDF.

Componentes:

Esta Ventana tiene dos paneles que llamaremos Frame superior, Frame intermedio, Frame inferior, en cada uno de ellos deben ir ciertos componentes gráficos de interfaz que detallo a continuación

(Ver video para más detalles)

1. Frame Superior: Visualización de Menús Disponibles

○ Tarjetas de Menús:

- En el frame superior se generan dinámicamente tarjetas (CTkFrame) que representan cada menú disponible. Cada tarjeta muestra una imagen representativa del menú (CTkImage), el nombre del menú y responde a eventos de click.

▪ Interactividad:

- Al hacer clic en una tarjeta, el menú correspondiente se añade al pedido actual, descontando automáticamente los ingredientes necesarios del stock. Si no hay suficiente stock para preparar el menú, se muestra un mensaje de advertencia mediante un mensaje (sugerencia usar CTkMessageBox para crear los mensajes)

2. Frame Intermedio: Gestión del Pedido y Resumen

○ Botón "Eliminar Menú":

- Situado en la parte superior derecha del Frame intermedio, este botón (CTkButton) permite al usuario eliminar un menú seleccionado del pedido. Al hacerlo, los ingredientes correspondientes se reponen en el stock y se actualiza el total del pedido.

○ Label de Total:

- A la derecha del botón "Eliminar Menú", un Label muestra el total acumulado del pedido. Este total se actualiza automáticamente cada

vez que se añade o elimina un menú del pedido, mostrando al usuario el costo total en tiempo real.

3. Frame Inferior: Detalle del Pedido

- **Treeview de Pedidos:**
 - En el frame inferior se encuentra un Treeview que muestra el detalle del pedido actual, incluyendo:
 - **Nombre del Menú:** El nombre del menú seleccionado.
 - **Cantidad:** La cantidad de veces que se ha añadido el menú al pedido.
 - **Precio Unitario:** El precio de venta por unidad de cada menú (descrito en la tabla de menú de más arriba).
 - Este Treeview permite al usuario ver un desglose claro de los menús seleccionados, facilitando la gestión del pedido.
- **Botón "Generar Boleta":**
 - Justo debajo del Treeview, se encuentra un botón (CTkButton) que permite generar una boleta en PDF para el pedido actual. Si no hay menús en el pedido, se muestra un mensaje de advertencia para evitar la generación de una boleta vacía. (sugerencia usar la librería **fpdf** de python)

Interacción con el Usuario:

- **Agregar Menús:**
 - Los usuarios pueden agregar menús al pedido haciendo clic en las tarjetas correspondientes. El sistema verifica automáticamente el stock disponible y descuenta los ingredientes necesarios si el menú puede prepararse.
- **Eliminar Menús:**
 - Al seleccionar un menú en el Treeview y presionar el botón "Eliminar Menú", el menú se elimina del pedido y los ingredientes se reponen en el stock.
- **Generar Boleta:**
 - Después de completar la selección de menús, el usuario puede generar una boleta que detalla los menús seleccionados, las cantidades y los precios totales.

Funcionalidad Adicional:

- **Actualización Automática:**
 - Cada vez que se selecciona la pestaña "Pedido", el sistema actualiza el stock disponible para asegurar que la información mostrada sea precisa.
- **Gestión Dinámica del Stock:**

- El stock se actualiza dinámicamente cada vez que se agrega o elimina un menú del pedido, asegurando que la información del pedido refleje con precisión los ingredientes disponibles (ver video)

Criterios de Evaluación

Criterio	Descripción	Excelente (10 pts)	Bueno (8 pts)	Suficiente (4 pts)	Insuficiente (3 pts)	Peso (%)
Implementación de la Orientación a Objetos	Uso correcto de clases, atributos, comportamiento de objetos.	Utiliza todos los conceptos de OO correctamente y de manera eficiente.	Utiliza la mayoría de los conceptos de OO correctamente .	Utiliza algunos conceptos de OO, pero con errores.	Uso incorrecto o inexistente de los conceptos de OO.	30%
Funcionalidad	El proyecto cumple con los requisitos funcionales especificados.	Cumple con todos los requisitos funcionales sin errores.	Cumple con la mayoría de los requisitos, con pocos errores.	Cumple con algunos requisitos, pero presenta varios errores.	No cumple con los requisitos funcionales especificados.	20%

Presentación Oral	Claridad y organización en la presentación, descripción de la problemática, diagramas, solución, demostración de código y manejo de preguntas.	Excelente presentación oral; describe la problemática y solución de manera clara, responde adecuadamente las preguntas y cumple con el tiempo de 15 minutos.	Buena presentación describe la problemática y solución de manera clara, aunque con detalles menores; responde adecuadamente las preguntas ;Excede el tiempo definido.	Presentación suficiente pero con falta de claridad y organización; la descripción de la problemática y solución es incompleta; manejo básico de preguntas ; Excede el tiempo definido.	Presentación pobremente organizada, falta de claridad y omisión de varios elementos clave; manejo ineficiente de preguntas; Excede el tiempo definido.	25%
Implementación de Interfaz Gráfica	Uso adecuado de una interfaz gráfica para mejorar la experiencia del usuario.	Interfaz gráfica bien diseñada, intuitiva y completamente funcional.	Interfaz gráfica funcional y adecuada, con pequeños detalles por mejorar.	Interfaz gráfica básica, con varios aspectos que podrían mejorarse.	Interfaz gráfica pobremente implementada o inexistente.	20%
Evaluación del Diagrama de Clases	Correcta representación de las clases, relaciones, herencia y otros elementos orientados a objetos en un diagrama de clases.	Diagrama de clases claro, completo y correctamente estructurado, mostrando todas las relaciones adecuadas.	Diagrama de clases funcional y correcto, con algunos detalles por mejorar.	Diagrama de clases con errores o ausencias en algunas relaciones.	Diagrama de clases incorrecto o incompleto, faltando elementos clave.	5%

¿Qué debo entregar?

Debes preparar una presentación **PPT en grupos de 2-5** para el día miércoles 09/09/2024 la cual debe contener:

- 1) Presentación del grupo y tema (título del problema, nombre de los integrantes, fecha y logo del departamento).
- 2) El diagrama de clases con la solución propuesta del problema.
- 3) Presentación de la solución en código explicando el flujo principal de la solución.
- 4) La presentación de la interfaz visual del problema.
- 5) La demostración de la funcionalidad total del programa.

La presentación tiene un tiempo máximo de 15 minutos y luego una tanda de preguntas de 5 minutos.