**ESTRUCTURA DEL PROYECTO FINAL POO (Boceto)**

**PROYECTO: Aplicación de sistema de inventario**

**CRITERIOS DEL PROFE (OBLIGATORIO):**

Condiciones:

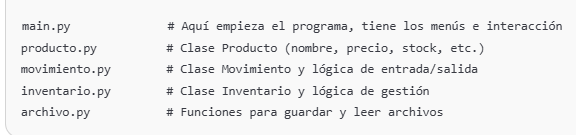
* Código original
* Uso de herramientas vistas en el curso ()
* Interacción y manejo a través de la consola (GUI opcional)
* Operaciones:
* Crear objeto a almacenar (+20, con diferentes atributos, el más importante es precio)
* Operaciones de registro de entrada, salida
* Método para obtener listado de inventario actual
* Manejo de fechas en los registros

**Características adicionales:**

* Carga masiva de registros
* Manejo de archivos para persistencia de datos
* Generación de informes en forma de documentos.

**MODULOS Y PAQUETES (Para tener en cuenta)**

. **Separar el código en módulos (.py)**

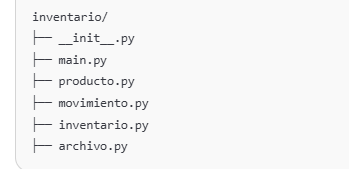
****

. **Usar import para reutilizar código**

****

. **Evitar código ejecutable fuera de main.py.** Los otros archivos (producto.py, archivo.py, etc.) solo deben tener **clases y funciones**, sin menús ni input(). Todo lo que interactúe con el usuario va en main.py.

. **Opcional, pero recomendado: agrupar en una carpeta (paquete)**

****

**PREGUNTAS CLAVE Y ESPECÍFICAS**

**Sobre el restaurante:**

1. ¿Cuál es el tipo de restaurante y su modelo de operación? (Ej. ¿autoservicio, mesa tradicional, cocina oculta?)
2. ¿Qué tipos de productos se gestionan en el inventario: ingredientes, suministros, productos finales, empaques, bebidas, ¿otros?
3. ¿Las recetas se estandarizan (fichas técnicas)? ¿Debe registrarse el consumo de ingredientes por preparación de cada plato?
4. ¿Hay variaciones en el menú o rotación de productos según temporada o día?
5. ¿Se deben registrar mermas, desperdicios o productos defectuosos?

**Sobre el inventario:**

1. ¿Se hace control por unidades físicas, unidades contables o ambas?
2. ¿Qué atributos debe tener cada producto? (Nombre, precio unitario, unidad de medida, proveedor, fecha de ingreso, fecha de vencimiento, código interno...)
3. ¿Los productos se organizan por familias o categorías contables? (Ej. cárnicos, lácteos, secos, bebidas...)
4. ¿Se gestionan inventarios por lote o por producto individual?
5. ¿Cómo se maneja el stock mínimo o punto de reposición? ¿Hay alertas automáticas?
6. ¿Se requiere registrar ubicación física del producto? (Ej. refrigerador 1, estantería 3)

**Sobre movimientos del inventario**

1. ¿Qué tipos de entradas existen? (Compras, devoluciones de platos no servidos, producción interna, correcciones por ajuste de inventario…)
2. ¿Qué tipos de salidas existen? (Preparación, vencimiento, robo/pérdida, donación, ajustes)
3. ¿Cada entrada/salida se vincula con una orden, usuario o documento soporte?
4. ¿Es necesario manejar trazabilidad: ¿quién hizo qué, cuándo y por qué?
5. ¿Se debe validar automáticamente si hay suficiente inventario antes de permitir una salida?

**Sobre Fecha y seguimiento**

1. ¿Qué fechas deben controlarse para cada producto? (Recepción, producción, caducidad, consumo)
2. ¿Se necesita un módulo de alertas por productos próximos a vencer o en exceso de tiempo en bodega?
3. ¿Hay necesidad de generar registros históricos o auditorías por fecha para trazabilidad?

**Sobre los usuarios del Sistema**

1. ¿Qué roles de usuario operan el sistema? (Ej. jefe de cocina, bodega, administrador, proveedor, mesero)
2. ¿Cada rol tiene un alcance diferente dentro del sistema? ¿Quién puede modificar, solo ver, eliminar registros?
3. ¿Debe quedar guardado el nombre o ID del usuario que realiza cada transacción?

**Sobre informes y reportes**

1. ¿Qué informes necesita el restaurante? (Inventario actual, rotación de productos, productos más consumidos, desperdicios, entradas/salidas por día)
2. ¿Qué periodicidad tienen los informes? (Diario, semanal, mensual)
3. ¿Qué formato se espera para los informes? (TXT, CSV, Excel, PDF)
4. ¿Los informes deben poder filtrarse por fecha, categoría, producto o responsable?

**Sobre carga máxima (Adicional)**

1. ¿Desde dónde se hace la carga masiva? (Archivo CSV, Excel, lectura de base de datos, formulario por consola)
2. ¿Qué campos mínimos debe tener un registro cargado masivamente?
3. ¿Se debe validar que los registros estén completos y sean coherentes antes de cargarlos?
4. ¿Qué sucede si hay errores en la carga masiva? (¿Se cancela todo, se omiten los errores, se hace log?)

**Sobre estructura y lógica del sistema**

1. ¿El sistema guarda la información en memoria temporal, archivos locales o una base de datos?
2. ¿Se debe implementar autenticación de usuarios?
3. ¿Se desea que el sistema tenga sesiones persistentes o reinicio diario?
4. ¿Qué opciones se mostrarán en el menú principal de consola? ¿Qué navegación tendrá?
5. ¿Las clases deben reflejar procesos operativos reales (compras, ventas, producción)?
6. ¿Se contempla el crecimiento del sistema a futuro (multi-sucursal, más usuarios, más módulos)?

**Sobre errores y validación**

1. ¿Qué pasa si se intenta registrar una salida de un producto que no existe o está agotado?
2. ¿Cómo manejar duplicados al ingresar productos (mismo nombre o código)?
3. ¿Qué controles se harán en la carga masiva? (Encabezados, formatos, campos vacíos…)
4. ¿Se requiere hacer logs de errores o simplemente mostrar mensajes?

**Sobre la interfaz de usuario**

1. ¿Qué menú principal se le presentará al usuario? (Ej. “1. Ver inventario, 2. Agregar producto…”)
2. ¿Cómo se asegura que la interacción sea clara para usuarios sin conocimientos técnicos?
3. ¿Deseas que el menú cambie según el tipo de usuario (ej. administrador vs. empleado)?
4. ¿Qué tipo de validaciones deben hacerse con los datos ingresados por consola? (Ej. evitar campos vacíos, validar fechas y números)
5. ¿Qué pasa si el usuario se equivoca o ingresa datos inválidos? ¿Se debe volver al menú o mostrar un mensaje de error?
6. ¿Se permite navegación anidada? (Ej. Submenús para tipos de productos)

**Sobre persistencia de datos (Adicional)**

1. ¿Qué datos deben guardarse permanentemente? (productos, registros, usuarios, logs…)
2. ¿Cómo se cargarán los datos cuando se abra la aplicación?
3. ¿Con qué frecuencia se guardarán los cambios? ¿Después de cada acción, o al salir?
4. ¿Deseas implementar copias de seguridad automáticas?

**Link de respuestas sugeridas:** [**file:///C:/Users/Camilo/Downloads/Respuestas%20sugeridas%20Proyecto%20POO.pdf**](file:///C:/Users/Camilo/Downloads/Respuestas%20sugeridas%20Proyecto%20POO.pdf)