**Informe proyecto final POO**

Este informe presenta el diseño, desarrollo e implementación de un sistema de gestión de pacientes y planes de dieta a través de una interfaz de consola destinada a nutricionistas. Se incluye una descripción detallada del proceso de diseño, los desafíos encontrados y sus soluciones, así como propuestas de mejora para el sistema.

**Proceso de Diseño:**

Inicialmente, se creó un modelo de objetos que incluye las clases: Dietplan, Dietitian, Meal, Patient y otras dos clases para manejar la funcionalidad de archivos CSV, llamadas DemoCsv y Main. El sistema fue configurado para que los nutricionistas se conectaran o registraran a través de menús pregrabados con las opciones a ejecutar. Sin embargo, un problema surgió cuando los menús redirigían al sistema a repetir la tarea de inicio de sesión o registro del nutricionista en lugar de comenzar con la gestión del paciente, lo que no era lógico según los objetivos planteados. Por ello, se reorganizó el código utilizando un bucle while que terminaba cuando el nutricionista se registraba o identificaba correctamente con su ID, o repetía en caso contrario.

Para recuperar los datos de cada paciente, el código se diseñó de manera que al introducir el ID del paciente en la consola, se mostrara toda la información relevante, incluyendo sus datos personales y su plan dietético. No obstante, se encontró un problema debido a que la ubicación de cada dato se almacenaba en un archivo CSV, dificultando la búsqueda del ID correspondiente.

**Problemas Encontrados y Soluciones**

**Problema 1:** Manejo de Entrada de Datos

**Descripción**: Las entradas incorrectas del usuario en la consola causaban errores en el programa.

**Solución**: Se implementó una lógica de validación para verificar las entradas del usuario y mostrar mensajes de error adecuados, guiando al usuario hacia entradas válidas.

**Problema 2:** Gestión de Actualizaciones de Información

**Descripción:** Actualizar información de pacientes o planes de dieta requería un proceso manual para buscar y actualizar los datos en los archivos CSV.

**Solución:** Se desarrolló un método para buscar y actualizar la información de manera más eficiente, utilizando estructuras de datos adecuadas.

**Propuestas de Mejora**

**Mejora en la Interfaz de Usuario**

Propuesta de una interfaz más intuitiva y amigable, con mensajes más descriptivos y opciones claras para los usuarios.

**Mejora en la Gestión de Datos**

Sugerencia de implementar una capa de persistencia de datos más robusta, como el uso de una base de datos en lugar de archivos CSV, para una gestión más eficiente y segura de la información.

**Añadir Funcionalidades Adicionales**

Recomendación de agregar funcionalidades como la búsqueda avanzada de pacientes, reportes de progreso y seguimiento de planes de dieta para mejorar la experiencia del usuario y la eficiencia del sistema.