**Título del Proyecto:**  
**Planificador de Comidas Semanal**

**Autor: Joel Alexander Jiménez Jiménez**

**Objetivo General:**  
Desarrollar una aplicación de consola que permita al usuario organizar su alimentación semanal, permitiendo agregar, visualizar, buscar, eliminar comidas y generar automáticamente un plan semanal. Este proyecto integra múltiples paradigmas de programación: imperativo, funcional y asincrónico.

**Descripción de la Aplicación:**  
El programa ofrece una interfaz en consola donde el usuario puede registrar comidas con sus respectivos ingredientes, generar un plan semanal de manera aleatoria, buscar comidas por ingrediente o nombre, eliminar comidas existentes, y guardar/cargar los datos. El estado se mantiene entre sesiones gracias al almacenamiento en archivos JSON.

**Uso de Paradigmas de Programación:**

* **Paradigma Imperativo:**  
  La estructura general del programa, incluyendo el menú principal, el flujo de control, la navegación entre opciones y la interacción secuencial con el usuario, está desarrollada de forma imperativa. Por ejemplo, el archivo main.py contiene funciones con instrucciones explícitas paso a paso (e.g., agregar\_comida, eliminar\_comida, ver\_plan, etc.).
* **Paradigma Funcional:**  
  En el archivo logic.py se implementan funciones puras que procesan datos sin causar efectos secundarios. Por ejemplo:
  + generar\_plan\_aleatorio(comidas) genera una nueva instancia del plan sin alterar las comidas originales.
  + buscar\_comidas\_por\_ingrediente y filtrar\_comidas\_por\_nombre utilizan filter y list comprehensions para transformar colecciones de forma declarativa.
* **Paradigma Asincrónico:**  
  El archivo storage.py hace uso del módulo aiofiles para implementar operaciones asincrónicas de lectura y escritura de archivos. Las funciones guardar\_comidas, cargar\_comidas, guardar\_plan y cargar\_plan son todas async, lo que permite una mejor simulación de operaciones concurrentes de E/S sin bloquear el flujo principal del programa.