APARTADO 4

# ¿Por qué Python?

La razón es la sencillez y capacidad de este programa para el análisis y tratamiento de los datos usando librerías como Pandas o Numpy, Base gran parte de la metodología de análisis usada en este proyecto en el libro: LIBRO.

Las distintas posibilidades que se barajaron antes de empezar fueron Java, Android, R y Python. Al final me decanté por Python por diferentes motivos, el primero en ser descartado fue Java, tras un estudio inicial sobre lo que quería hacer y el como hacerlo, me di cuenta que necesitaba un lenguaje potente que me dejase tratar los datos con claridad para su análisis, y después de leer varios artículos como: REF\_ARTICULOS, me percaté de que Java no era la mejor opción, estos artículos siempre te orientaban hacia Python y R. Entonces, ¿Por qué Android?, por la sencilla razón de que este trabajo está desarrollado para unir el mundo de la dietoterapia, y las ciencias de la Salud con la informática para hacer que el usuario tenga un fácil aprendizaje de dicha metodología. A mi parecer la forma más clara y rápida de llevar al usuario dicha tecnología es a través de su SmartPhone, pero se acabó descartando debido a la falta de conocimientos sobre sistemas Android. A estas alturas ya solo me quedaba elegir entre Python y R, tras indagar a superficialmente sobre ambos lenguajes para el análisis y el tratamiento de datos, se llego a la conclusión que ambos lenguajes tienen una forma de trabajar muy similares, y entonces la decisión fue clara, debido a que he trabajado en numerosas ocasiones con Python y la sintaxis de R para mi era desconocida, al final, me decante por Python.

# Metodología

## Introducción

En este subapartado explicaremos el cómo y porque se ha tratado los datos en este TFG además de los diferentes cálculos internos que se realizan para el sistema de recomendaciones, cálculos, etc.

## Excel y Pandas

Se usará la herramienta de Microsoft, Excel, para trabajar como una base de datos, y poder tratarlos en forma de DataFrame, esta particularidad, nos la da la librería de Pandas. Hay un Excel para la base de datos genérica, y otro para la base de históricos. Se leen los datos automáticamente en cuanto el usuario entra en la aplicación, pero solo se guarda si el usuario así lo desea.

## DataFrame

Se usan los DataFrame, para llevar un registro de todos los datos que el programa necesita, se tiene en cuenta tanto las bases de datos de los alimentos, usuarios y comidas como lo que el usuario lleva en el día.

Además en el momento en el que se registra el usuario se separan las comidas de la lista principal, creando 5 listas (Desayuno, Merienda, Comida, Almuerzo y Cena), se tratan por separado y se filtran por el umbral, el umbral es un medidor de calidad de la comida, para qué solo te muestre lo más sano en un principio, y se van actualizando sensibles a los cambios de la aplicación. Si se refrescar aumenta el umbral para que sea mas amplia las posibilidades de comida, aunque esto signifique una peor alimentación. Con cada elección se actualizan el resto de las comidas, para que se ajuste al máximo la dieta a sus necesidades, se ordena por la formula que se hablará mas adelante, se crea la variable “dif” que sale de esta fórmula, y este es el eje principal de las recomendaciones, cada vez que se elige una comida, la comida se almacena para poder tratarla mas tarde, en cuanto se selecciona los botones se desactivan, y se cambia el botón seleccionar a editar. Cada vez que se elige una comida se reparte las calorías sobrantes/que faltan, y se recalculan las posibilidades.

# Librerías

## Pandas

La librería desarrollada por Wes McKinney, Pandas, es una librería usada para el tratamiento de los datos como estructuras. No fue la primera opción para este proyecto, debido a que este proyecto se enfoca en el análisis de datos, y no en el tratamiento de bases de datos, se decidió crea una base de datos local en un archivo Excel, la primera opción fue la librería “openpyxl”, la cual, es una librería de código abierto, que permite la carga y manejo de datos XLS, el problema vino que te devolvía objetos desestructurados difíciles de tratar, y complicaba él objetivo principal del proyecto, acto seguido basándome en él libro: LIBRO, decidí probar con Pandas, esta librería me permitía leer los datos de los archivos .XLS, en forma de dataframes. Esto simplificaba el análisis de los datos, permitiendo tratarles tanto como dataFrame como en forma de numpy.array, dicha librería será analizada a continuación.

## Numpy

Como ya he nombrado en varias ocasiones en este memoria, el libro: LIBROO, se basa principalmente en esta librería, como añadido, es la librería sobre la que he trabajado a lo largo de la carrera en cuanto a análisis de datos en Python, y la mas recomendada por los usuarios en la web, sobre esta librería se sustenta principalmente este proyecto, siendo la encargada, del tratamiento, procesamiento y calculo, de todos los datos que internamente realiza el programa.

## Interfaz gráfica

La librería utilizada para realizar y diseñar la interfaz gráfica fue Tkinter, tras buscar en diferentes páginas que encontraremos en la bibliografía, me decante por Tkinter debido a la falta de conocimientos en interfaces gráficas con Python, me decante por la mas fácil de aprender. Entre las opciones que baraje se encontraban: Tkinter, WxPython, PyQt y PyGTK.

Tkinter traía una serie de ventajas, entre ellas que viene preinstalada con Python, era fácil de aprender, y hay una documentación amplia y extensa. Pero también tiene sus desventajas, incluye pocos elementos gráficos, tiene un limitado control y la navegabilidad mas sencilla se hace complicada, y sobre todo lo que mas noto es su lentitud, cuanto mas elementos añadas, mas lento va la interfaz, no tiene alguna especie de cache o memoria que guarde lo que has “dibujado”, sino que los dibuja cada vez que salen en pantalla, lo hace sobre cada botón etiqueta, etc.

La segunda opción que se me paso por la cabeza, fue WxPython, tenia grandes ventajas, como la rapidez, la flexibilidad que este ofrecía y sobre todo su mayor ventaja son todas las opciones que tiene para crear una interfaz gráfica compleja y “profesional”, tras barajar, estas opciones y compararlas con las de Tkinter, me di cuenta que todo lo que te ofrecía WxPython, era innecesario para la interfaz tan simplista que necesitaba para mi proyecto, y el aprendizaje era más complejo, además de ser mas complicado encontrar información o documentación sobre esta librería. El mayor problema de esta librería era que tiene una comunidad muy activa, la cual, esta constantemente insertando cambios, esto era un problema para proyectos largos, pues creaba problemas de compatibilidad, pero para este proyecto, no era algo que me resultará un inconveniente.

El resto de librerías fueron descartadas, al poco de buscar información sobre ellas debido a que daban las mismas ventajas o similares que la WxPython, pero tenían mas inconvenientes, al menos para el tema que aborda este proyecto.

# Módulos

## CalculosDieta

### Introducción

Es el modulo principal sobre el que se sustenta el programa, hace todos los cálculos de recomendación, kilocalorías, repartos, etc.

En el manual del programador se extenderá mas sobre cada función, a continuación, se hará un breve comentario sobre cada función en base a la metodología.

### CalculoTMB, rapartoDeKcal y distribuciónDeMacronutrientes

Estas tres funciones trabajan de manera paralela, el **calculoTMB**, lo que hace es calcular el gasto calórico basal, de aquí se saca el objetivo diario, una vez sacadas la cantidad de Kilocalorías diarias, se reparten las kcal (**repartoDeKcal** )según sea desayuno, almuerzo, merienda o cena. Y a la par se calcula el número de Kilocalorías que se tiene que tomar de hidratos, grasas y proteínas, en base al tipo de dieta recomendable para tu patología en la función distribuciónDeMacronutrientes.