ASPECTOS RELEVANTES

# La idea

Uno de los aspectos a destacar en el desarrollo de este proyecto, es el nacimiento de dicho proyecto. Pues se trata de un trabajo de fin de grado, el cual, es un trabajo puro del alumno, donde tanto idea como desarrollo son realizados por él, y gran parte del tiempo gastado en el desarrollo fue en la ocurrencia de varias ideas, el pulimiento de dichas ideas, y la conversión de todo esto en lo que se convirtió en la idea final. Aunque dicha idea fuera cambiando a lo largo de su desarrollo.

Todo llego de la necesidad de crear un trabajo interesante, algo diferente y que fuera de actualidad, a esto se le sumo, que para un desarrollo mas ágil debía ser un tema conocido e interesante para su desarrollo. Entonces se llevo a cabo un periodo de investigación de las necesidades que azotan a las personas hoy en día, con las ultimas corriente de moda y que sobre todo pudiera resultar útil. Tras varios días se llegó a la conclusión de que actualmente la gente tiene una constante necesidad por cuidarse, por la vida sana, y que, además de ser una necesidad básica para cualquier persona, se estaba tornando en una moda explotable en cualquier página web, red social, etc.

Una vez encontrado el groso del proyecto tocaba perfilar detalles, para ver a través de que medio se podía llevar “una mejor calidad de vida” al usuario. De aquí surgieron varias vertientes:

* Deporte: descartada, por la inmensa cantidad de aplicaciones de planificación de entrenamiento que existes a día de hoy.
* Nutrición: Inicialmente descartada, debido a las mismas razones que lo anterior, existen miles de aplicaciones tanto que te proponen una dieta como que te ayudan a seguirla.
* Recordatorios: Con esto me refiero, a recordatorios de cuando hacer deporte, de cuando tomarse los medicamentes, etc. Se descarto rápidamente por ser una más, a la par que un proyecto banal.
* Nutrición con otro enfoque: Aquí es donde llego la idea, una vez descartada estas tres, se pensó que estaba todo inventado y el proyecto se convertiría en otra forma de “reinventar la rueda”, hasta que nos percatamos, que no existe una aplicación que te enseñe a comer, y que tenga en cuenta las patologías aunque sean en menor medida. Ahí es cuando nació la idea, y se empezó a formar la idea de la estructura de lo que iba a ser el proyecto.

De este modo se empezó con el proyecto, partiendo totalmente de cero, hasta el punto de conseguir algo perfectamente funcional, cumplimentando los objetivos propuestos para dicho proyecto.

# Formato de trabajo

Se desarrollo una metodología ágil de forma incremental, se empezó formalizando las bases hasta llegar a la etapa final. No se pasaba a la siguiente etapa sin haber terminado la anterior.

Las tres principales etapas del proyecto fueron:

1. Funcionamiento por interfaz Shell
2. Funcionamiento de manera gráfica
3. Finalización de cada detalle dejado indicado durante el proyecto.

De esta forma, se fueron creando todas las características de manera incremental, de forma que se dio mas importancia al claro desarrollo del proyecto, que a la velocidad de dicho desarrollo. Pese a esta importante decisión, en ningún momento el proyecto se atasco, y se pudo llevar a cabo de manera ágil, gracias a la reutilización de código entre capas, permitiendo una clara transparencia a la hora del desarrollo.

Al no lanzarnos directamente con el entorno gráfico, el cual, nunca había sido visto a lo largo de la carrera, permitió que fueran más obvios los pequeños errores de implementación, de cálculos, etc. Para que cuando se pasará a la etapa de desarrollo gráfico, nos pudiéramos centrar en los errores, o correcciones que el entorno gráfico suponía, sin amontar pequeños fallos, que mas tarde dieran lugar a grandes errores.

# Algoritmos del programa

Se quiere hacer énfasis en dichos algoritmos debido a que el programa utiliza una serie de algoritmos propios, ideados por el autor, que permite que el sistema de recomendación funcione como es debido. En este subapartado no se va a extender con la explicación de cada uno de los algoritmos utilizados, o métodos similares, creados por terceros y desarrollados por el alumno, sino aquellos que han sido ideados por él, y son un aspecto relevante en el desarrollo por el tiempo de ideación que han llevado dichos.

## Algoritmo de Recomendación

Es el algoritmo que estructura todo el sistema de programación, nace de la necesidad de que la recomendación se ajuste al máximo a las necesidades del usuario. Si se ordenaba única y exclusivamente por la calidad, nunca se verían los menús de peor calidad, además de que podría haber una desigualdad inmensa entre las calorías ingeridas y las necesitadas. Mas tarde, se pensó en hacer una formula basada en la diferencia calórica, cosa que se adecuaba mas a las necesidades del usuario, pero creaba un desajuste en los macronutrientes, pues el usuario podía llevar 300 kcal exclusivamente de grasa, y que le recomendase otras 300 también solo de grasa, rompiendo la idea de dieta equilibrada. Por ello, se ideo otro prototipo de formula.

La formula tiene en cuenta la proporción de las necesidades calóricas del usuario, partiendo de cada macronutriente, se calcula la diferencia entre la diferencia de macronutrientes que debería llevar y los que llevo, todo llevo partido por la suma de la diferencia entre: Las kilocalorías que debo comer y las que tiene el alimento y la diferencia entre el macronutriente especifico y lo que el alimento tiene.

**FORMULA**

## Persistencia base de datos

Esto en verdad, no es implementado como tal por el alumno, sino más bien, es un recurso para romper una de las limitaciones que presentó utilizar Excel como simulación de la base de datos.

Para evitar datos sobre escritos y problemas con los datos, se procuró durante todo el programa atomizar lo máximo posible cada método sobre la base de datos, haciendo que los cambios sean lo mas inconexos y manejables para el usuario.

En el método añadir alimento se rompe esta norma, añadiendo un alimento en la base de datos que hay que guardar, pero sin guardar todos los cambios en el resto de los alimentos, por si el usuario no quisiera hacerlo. Para evitar esta falta de control por parte del usuario sobre la persistencia de sus datos, a la hora de almacenarlo, se carga el array de nuevo, y se añade esta nueva línea tanto al array usado durante el programa, como al nuevo array de alimentos, este nuevo array se almacena en la base de datos, mientras que el otro sigue con su propósito en el programa. Si el usuario hace algún cambio y lo guarda, el nuevo alimento estará a salvo, si, por el contrario, el usuario decide salir sin guardar su progreso en el día, el nuevo alimento ya estará guardado en la base de datos.

## Distinción entre tipos de comida

Uno de los retos a afrontar a lo largo de este proyecto fue simplemente, el como, se iba a indicar que tipo de comida/menú era. Sin duda no fue uno de los retos más complejos abordados a lo largo del proyecto, pero si uno de los más curiosos.

Recordamos que un mismo menú, puede pertenecer a varias comidas del día, entonces se pensaron varios métodos hasta llegar al óptimo, los cuales explicaremos con brevedad a continuación:

En un principio se planteo hacer una cadena de Strings separadas por algún delimitador cualquiera, y cargarlo en forma de String y separarlo por algún método (Por ejemplo Slice()), pero dio ciertos problemas de carga y almacenamiento, problemas no demasiados complejos, pero que aumentaban la complejidad algoritmo al buscar a través de un bucle todas las posibilidades complicando esto, y se pensó en una concatenación de bits donde 1 es que era esa comida y 0 que no, es decir: 11010, Significa que no es ni comida ni cena. Esto presentaba el mismo problema que la cadena de caracteres y era quizás más confuso para su compresión, Además del pequeño inconveniente de que si un menú no era desayuno ( Es decir empezaba en 0), en algunas ocasiones se almacenaba sin ese digito creando una desigualdad en el programa. Por estas razones se deicidio pasar al siguiente método, una mezcla de ambas, se iba a usar una cadena para calcular un único valor que cargar. Finalmente se paso esa cadena de bits a un valor numérico, de esta manera se asegura una única carga, y el tratamiento a través de cláusulas condicionales simples.

Ejemplo:

Un alimento que sea comida y cena: 00101 -> Se representará con el número 5.

Esto permite la carga de un único elemento, sin posibilidad de fallo además de un calculo sencillo, cuyo mayor inconveniente es llevar a cabo todas las clausulas condicionales, para hacer la criba por tipo de comida.

# SCRUM

Quizás uno de los aspectos más destacados a la hora de realizar el proyecto, fue la organización por parte del alumno, a través de la metodología SCRUM, que permitía al alumno mantenerse centrado en cada nueva tarea, no pasando a la siguiente hasta dejar esta zanjada del todo.

Para la realización de esta metodología se tuvo en posesión una pizarra, en la cual se iba pegando post-it con una breve descripción de la tarea y una fecha máxima para la realización de esta. La pizarra era dividida en cuatro partes (pequeña adaptación del SCRUM).

1. TODO – Por hacer, eran todas las tareas, pautas, e ideas que iban surgiendo a lo largo de la semana o tras las reuniones semanales. Había un máximo de cinco, para evitar que el proyecto se atascase por una carga excesiva de trabajo. De TODO se pasaba a in progress.
2. IN-PROGRESS: En proceso, máximo dos cosas, una principal y una secundaria, la principal, consistía en el desarrollo de grandes métodos, cálculos, etc. Mientras que los secundarios, podían ser trabajos de investigación, rellenar datos o cualquier detalle que faltase de completas. De esta manera se evitaba la bifurcación del trabajo, haciendo del camino, pequeños objetivos, claros y directos a completar. Cuando el alumno pensaba que había terminado la tarea, la movía a “TO-VERIFY”.
3. TO-VERIFY: A verificar, es decir, falta la aprobación de los cambios realizados, si eran cambios propuestos en las reuniones semanales, eran zanjados en la siguiente reunión semanal; De esta forma se creaban distintos modelos y versiones de métodos, perfilando al máximo cada detalle en cada método. Las pequeñas tareas, eran resueltas, ya fuera en las reuniones semanales, o preguntando a personas, sobre como sería para ellos el correcto funcionamiento del programa.
4. DONE: Una vez todas las partes se ponían de acuerdo en que la tarea estaba terminada, pasaban a DONE, este apartado existía básicamente, para hacer mella en el trabajo que el alumno realizó durante la semana, haciendo palpable las semanas donde se tuvo una mayor o menor carga de trabajo, pudiendo regular esto, y teniendo de esta manera un avance constante.

En progreso solo se podía encontrar una tarea, y si el tutor desechaba una tarea como incompleta pasaba a todo de manera que, de esta forma, semanalmente se cumplían con una serie de propósitos de forma clara y concisa, y no se apelotonaban todas las ideas y propuestas dejando pequeños cabos sueltos que arruinaran la representación final del proyecto.

Este método fue estudiado en la asignatura de gestión de proyectos por el alumno, y culminado en su periodo de practicas con el instituto de castilla y león, el cual, usaban de manera muy similar a la adaptación del alumno; La metodología SCRUM para gestionar todos sus proyectos.

# Conocimientos aprendidos en la carrera

## INTRODUCCIÓN

Apartado donde se hablará de forma breve sobre todo conocimiento aprendido durante los años de carrera, relacionado directamente con el desarrollo del proyecto.

## Conocimientos sobre Python

A lo largo de la carrera, se han estudiado diferentes lenguajes de programación (Ej: Java, C, C#...), entre los cuales se encontraba Python. Por ejemplo en asignaturas como Sistemas Inteligentes, Nuevas tecnologías, etc., o en mi paso por la politécnica de Varsovia, en asignaturas como “Algoritmia y computación matemática”, la cual, toda la asignatura consiste en un gran proyecto, donde se ponía a prueba los conocimientos algorítmicos en el lenguaje Python.

Además, multitud de las librerías usadas durante este proyecto fueron utilizadas en dichas asignaturas, haciendo que el proyecto, partiera desde un punto avanzado facilitándome el aprendizaje.

## Conocimientos sobre la gestión de proyectos

A fin de cuentas, la metodología que se ha usado para idear, planificar y desarrollar el proyecto, es el conjunto y suma de una serie de actitudes adquiridas a lo largo de los últimos años. Ya sea con asignaturas dedicadas específicamente para ello como fue Gestión de proyectos, como con el prueba y error, que ha hecho que se fuese aprendiendo la mejor manera de trabajar a nivel personal. Permitiendo que este proyecto, fuese con diferencia el mas amplio y hasta la fecha, mas cómodo, a la hora de ser elaborado.

Metodologías como el SCRUM, Las pruebas de errores, la organización interna del scrum, las reuniones semanales sobre el estado del proyecto, etc.

## Investigación

Básico, pero de vital importancia, es el nivel de capacidad de investigación y búsqueda de la información no es el mismo que al inicio del grado. La velocidad y capacidad, de saber buscar, es uno de los conocimientos más importantes que se adquieren a lo largo de la carrera. Conocimiento indirecto, pero presente en todas y cada una de las asignaturas que se han cursado, dicho nivel de investigación ha permitido que errores que antiguamente hubieran durado días, durasen horas, buscando en diferentes páginas y libros y contrastando cada información recibida a lo largo de este proceso de investigación

# Conocimientos externos a la carrera

## INTRODUCCIÓN

En este apartado, se explicarán los aspectos relevantes del desarrollo del proyecto externos a los conocimientos iniciales del alumno. Todo aquello que ha tenido que aprender para poder llegar al resultado final.

## PSICOLOGIA

Siendo sinceros, sería una parte de psicología, intercalada con otra parte de educación primaria. Para el desarrollo del proyecto, y el moldeado de lo que es la idea genérica de este; Ha sido necesario el estudio de ciertos campos para entender cual podría ser el mejor camino para el entendimiento con el usuario. Este proyecto no buscaba una forma rápida de llegar al cliente/usuario, sino constante, que el cliente convierta los cimientos de la aplicación, en su propio estilo de vida. Por ello se busco consejo a través de varias entrevistas a alumnos y graduados de psicología y magisterio, tras las cuales se llego al método: Aprender a aprender.

### Aprender a aprender

Resulta ser uno de los mejores métodos de aprendizaje ausente que hay. Donde el aprendizaje ausente, como su propio nombre indica, consiste en el aprendizaje ausente de figura dedicada a la enseñanza, para poder llevar a cabo ese proceso evolutivo. Aquí el profesor, sería la aplicación, pero al no tener conciencia propia, ni libre capacidad de elección, el método de enseñanza habitual queda descartado.

La idea principal nace de las ventajas que tiene implementar un estilo de vida, sobre el seguir ordenes estrictas. Tras varias entrevistas, se decanto por un estilo incremental, en el que se intenta concienciar al usuario de cada decisión que toma, haciendo que el aprendizaje se base en pequeñas metas personales, y en la concienciación del usuario para que el mismo se de cuenta de que es lo mejor para él y aunque de manera lenta pero segura, llegue a la meta que tenga como objetivo.

## NUTRICION

Se tenían conocimientos básicos sobre el mundo nutricional y el movimiento “real-fooding” o comida real, que consiste en comer no solo las calorías necesarias para subir o bajar de peso, sino comida saludable; A efectos prácticos, esa es la idea principal sobre la que se fundamente el proyecto. Buscar la manera, de que el usuario, adquiera una mejor calidad de vida, a través de comida saludable, y que ya de paso, consiga sus objetivos principales.

### “Calidad” de los alimentos

Es una parte fundamental del proyecto, la calidad marca el umbral por el cual se cribaran los alimentos y se aconsejarán principalmente los alimentos de mejor calidad, y esto irá variando conforme el usuario busque nuevos alimentos, u opciones.

## Nutriscore

En un principio para que la calidad fuera lo mas precisa posible, y pensando en esta aplicación como proyecto de negocio, se pensó que, un nutricionista, experto en la materia, fuera quien añadiera los menús y la calidad de dichos. Esto a nivel de negocio, era la idea más optima, pero a nivel de proyecto, dejaba un vacío en la clara objetividad en la selección de calidad y en la limitación de la base de datos. Por esta razón se eligió el algoritmo del semáforo o NUTRISCORE, algoritmo que se basa en una serie de características buenas y malas de los alimentos, y según el resultado de la diferencia, la cual, se acaba cribando en uno de los cinco posibles valores: A,B,C,D Y E, ordenador de mejor a peor.

**IMAGEN**

De esta manera conseguimos que el usuario pueda añadir sus propios alimentos y no haga “trampa” con la calidad de la comida. Manteniendo la integridad de la aplicación lo máximo posible.

### Lagunas en el algoritmo

Este algoritmo no es perfecto, pues se basa única y exclusivamente en cifras, en las cifras de azúcar, de proteína, de sal, etc. Por lo cual, cosas saludables como puede ser el aceite de oliva virgen extra, tienen como resultado una pésima calidad, mientras que, quizás, ultraprocesados como cereales bajos en azucares, bebidas edulcoradas, y cosas similares.

Esto se debe por ejemplo, a que por cada 100 gramos, el aceite de oliva tiene una alto contenido en kilojulios y en grasas, por lo cual lo coloca en la categoría E, cuando debería ser un A.

## INTERFACES GRAFICAS

Pese a tener los conocimientos básicos sobre programación en Python, se desconocía cualquier tipo de implementación de interfaz gráfica. Debido a esto, gran parte del inicio del desarrollo del proceso, fue la selección de la librería que se utilizaría durante el resto del proyecto para la implementación de este.

Finalmente se escogió Tkinter, por su sencillez, pues permitió dar grandes avances y adaptarse a este nuevo sistema rápidamente. Permitiendo que la interfaz se complicará de manera exponencial, aplicando todo aquello que se iba aprendiendo de manera colateral a lo ya existente.

# Fases del proyecto

Las fases de este proyecto pueden ser divididas de diferentes modos, según que criterio se utilice, de por si podemos dividir todo el proyecto en tres grandes fases: Idea, desarrollo y testing. Teniendo en cuenta esto, hay que tener en cuenta que el desarrollo a su vez, se dividió en otras tres fases principales que fueron: creación de estructuras iniciales, desarrollo en interfaz de comandos y desarrollo en interfaz gráfica.

Por lo cual un breve esquema del desarrollo podría ser:

* Nacimiento y pulimiento de la idea
* Desarrollo:
  + Creación de la estructura de los datos.
  + Carga y manejo de los datos
  + Creación del proyecto por interfaz de comandos
  + Traducción del proyecto a interfaz gráfica
  + Pulimiento de detalles y nuevas funcionalidades
* Pruebas
* Desarrollo de la documentación pertinente para plasmar dicho proceso

## Cronograma