



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

**Aplicación de recomendación
nutricional - HEALTHAPP**



Presentado por Jesús Carro Tomé
en Universidad de Burgos — 28 de junio
de 2019

Tutor: Antonio J. Canepa Oneto



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Antonio J. Canepa, profesor del departamento de Ingeniería Civil, área de Lenguajes y sistemas informáticos.

Expone:

Que el alumno D. Jesús Carro Tomé, con DNI 71308125Y, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado Aplicación de recomendación nutricional - HEALTHAPP .

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 28 de junio de 2019

Vº. Bº. del Tutor:

D. Antonio J. Canepa Oneto

Resumen

Una de cada cinco muertes en el mundo se debe a la mala alimentación, y a las diferentes patologías que se agravan con esta. Por ello, cada vez son más los profesionales dedicados al ámbito de la salud y la nutrición, y cada vez es mayor la presencia de dichos profesionales en centros sanitarios y hospitales.

Existen numerosas aplicaciones para la gestión de dietas y ejercicios, y las que existen para la enseñanza a través de métodos multimedia. Entre lo que existe, hay una carencia en sistemas de recomendación que ayuden a las personas con conocimientos mínimos; a adecuar ese estilo de vida saludable y adquirir dichas competencias.

Gran parte de nuestra calidad de vida depende de la motivación personal, del logro de conseguir nuevas metas, de la sensación de avanzar en aquello que nos proponemos. Por ello, el autoaprendizaje es usado en varios ámbitos como método de enseñanza, debido a que el sujeto aprende en base a sus errores y logros, lo que produce una ambición de mejorar y seguir adelante.

Por ello, a través de un sistema de recomendación y usando los algoritmos más novedosos en cuanto a la categorización de alimentos, este proyecto cumple con la necesidad de una aplicación que a través de ese aprendizaje, nos ayuden a crear las metas necesarias para con el tiempo lograr el objetivo de adecuar un nuevo hábito saludable de vida.

Descriptores

Dietoterapia, auto-aprendizaje, Python, Tkinter, Nutriscore, Nutrición, Sistema de recomendación, investigación.

Abstract

One in five deaths in the world is due to poor diet, and the different pathologies that are aggravated by it. Therefore, more and more professionals are dedicated to the field of health and nutrition, and the increasing presence of these professionals in health centers and hospitals.

Among the many applications that exist for the management of diets and exercises, and those that exist for teaching through multimedia methods. Among what exists, there is a lack of recommendation systems that help people with minimal knowledge; to adapt that healthy lifestyle and acquire those skills.

Much of our quality of life depends on personal motivation, achievement to achieve new goals, a sense of progress in what we set out to do. Therefore, self-learning is used in various fields as a teaching method, because the subject learns based on his mistakes and achievements, which produces an ambition to improve and move forward.

Therefore, through a recommendation system, and using the latest algorithms in terms of food categorization, this project meets the need for an application that, through this learning, will help us to meet the necessary goals, for the to achieve the goal of adapting a new healthy living room.

Keywords

Diet therapy, self-learning, Python, Tkinter, Nutriscore, Recommendation system, investigation.

Índice general

Índice general	III
Índice de figuras	V
Índice de tablas	VI
Introducción	1
Objetivos del proyecto	5
2.1. Objetivos generales	5
2.2. Objetivos Funcionales	5
2.3. Objetivos incrementales del desarrollo	6
Conceptos teóricos	9
3.1. Introducción	9
3.2. Nutrición	9
3.3. Prevenciones	13
3.4. Enseñanza	14
3.5. Autoaprendizaje del valor TMB	15
3.6. Métodos de recomendación	16
Técnicas y herramientas	17
4.1. ¿Por qué Python?	17
4.2. Metodología	18
4.3. Técnicas	18
4.4. Estructura del programa	23
4.5. Librerías	24

4.6. Módulos	26
4.7. Herramientas	34
Aspectos relevantes del desarrollo del proyecto	37
5.1. La idea	37
5.2. Formato de trabajo	38
5.3. Algoritmos del programa	38
5.4. Conocimientos aprendidos en la carrera	41
5.5. Conocimientos externos a la carrera	43
5.6. Fases del proyecto	45
Trabajos relacionados	47
6.1. Estudios:	47
6.2. Aplicaciones Similares	50
Conclusiones y Líneas de trabajo futuras	53
7.1. Conclusiones	53
7.2. Líneas de trabajo futuras	54
Bibliografía	57

Índice de figuras

3.1. Algoritmo Nutriscore	12
4.2. Ventana emergente de información	21
4.3. Ventana emergente que avisa que una acción no se ha podido realizar.	21
4.4. Ventana emergente que muestra que ha ocurrido un Error. . . .	22
4.5. Estructura base de datos alimentos	29
4.6. Estructura base de datos Usuarios	30
4.7. Estructura base de datos Patologías	31
4.8. Estructura base de datos Registro	32
5.9. Tabla cronograma al completo	46

Índice de tablas

3.1. Distintos tipos de dietas y su distribución	11
3.2. Distribución calorica a lo largo del día.	11
3.3. Resultado final del Algoritmo Nutriscore	13
4.4. Valor de la actividad física en TMB	20
4.5. Porcentajes totales de cada comida	27

Introducción

Actualmente se estima que alrededor del 81 por ciento de los españoles, que se proponen cambiar de hábitos y seguir una dieta acaba desistiendo [14]. Eso en su mayoría nace de la dificultad de mantener un ritmo constante de rigor a la hora de seguir una serie de instrucciones, que de una manera u otra te condicionan impidiéndote hacer una vida cien por cien normal, más allá de lo que en el ámbito alimenticio se refiere, como es en el ámbito social.

Por ello, combinando el análisis y tratamiento de datos mediante sistemas informáticos con los últimos estudios relacionados con la dietoterapia, se desarrolla este proyecto. No inculca una dieta estricta al usuario, sino que sirve de guía para percibir un nuevo estilo de vida como propio. Gracias a esto, se puede llegar a conseguir un gran avance médico/informático, debido a la importancia que la dietoterapia tiene en la salud de las personas.

El estudio *Global Burden of Disease* en *The Lancet*, que ha analizado las tendencias de consumo de quince factores dietéticos entre 1990 y 2017 en 195 países, señala que una de cada cinco muertes en el mundo está relacionada con la mala alimentación, sin considerar las muertes por desnutrición [12]. Por lo tanto, el desarrollo se sustenta en la constante necesidad y la creciente demanda de sistemas que ayuden y faciliten a una adecuada alimentación o dietoterapia para el cuidado personal.

Actualmente, la tecnología multimedia se abre paso en el ámbito educativo siendo un método muy explotado para el aprendizaje, llegando incluso a ser una asignatura del Grado de Magisterio en diversas universidades de España [9].

El estilo de aprendizaje autoaprendizaje, es más efectivo cuando reconoces que es lo más importante para ti" [3]

Tras estudiar varios métodos de aprendizaje, se ha considerado que cuidar la salud a través de la alimentación no se basa exclusivamente en una dieta, sino en un estilo de vida, y el autoaprendizaje sería el mas oportuno. A un ritmo lento y constante, el usuario aprende a comer bien y no se pierde entre estrictas dietas.

Se pensaron en una serie de pasos imprescindibles para inculcar un nuevo estilo de vida y su relación con el autoaprendizaje:

1. Identificar que se quiere cambiar.
2. Metas específicas, realistas y constantes.
3. Creación de un plan.
4. Recordatorios que seguir.
5. Mide tus avances.

Se ha logrado que la herramienta cree una sensación de falsa simplicidad, es decir, se ha buscado hacerlo visualmente más simple, para abordar cuestiones complejas. De esta manera y como veremos a continuación, los puntos anteriores los cuales son cimientos sobre los se ha basado el autoaprendizaje que imparte la aplicación, se cumplen en los puntos siguientes, a través de las funcionalidades ya implementadas.

1. Función principal, aprender y cambiar el estilo de vida a través de la dietoterapia, cumpliendo el objetivo uno de identificar que parte de tu estilo de vida deseas cambiar.
2. Una de las partes mas complicadas e invisibles del proyecto. El programa siempre recomienda la mejor opción, y lo realiza en base a los datos del usuario, consiguiendo que las metas sean constantes y que poco a poco se vayan volviendo mas estrictas, de forma que pase de manera imperceptible para el usuario.

3. Respecto a la creación de un plan, aunque siempre condicionado por las recomendaciones del programa, es el usuario el que lo elige. Estas recomendaciones le ayudan a tomar la decisión correcta, ya que cada opción que decide el usuario dentro de la aplicación es registrada, y se refleja en los gráficos de calidad permitiéndole ver en todo momento la conveniencia de sus elecciones.
4. A través del Historial y los gráficos de avance, el usuario puede ser consciente de su progreso. Esta parte es muy importante para motivar al usuario y que siga adelante con el plan. Dentro de estas opciones aparecen reflejados los puntos 4 y 5.

A lo largo de esta memoria será plausible el cómo se ha desarrollado cada parte del proyecto, tanto en el ámbito informático como en el nutricional. Detallando cada estudio, desarrollo y opción que se ha tenido a lo largo de este, para que toda decisión tomada a lo largo del periodo de desarrollo quede clara, y se entienda el por qué de cada decisión. Con este proyecto se consigue proyectar la unión de dos mundos complejos, de la manera más simple posible, haciendo alarde de que en la sencillez se oculta toda su complejidad.

Objetivos del proyecto

En este apartado hablaremos sobre los objetivos generales y específicos de este proyecto, aclarando las dudas sobre las intenciones de una de sus partes.

2.1. Objetivos generales

Desarrollar un Software que ponga a disposición del usuario, un sistema automatizado de planificación alimenticia, que mecanice y facilite el autoaprendizaje. De esta manera, será visible al usuario cómo influye cada decisión que toma en su día a día. Se da la misma importancia a cada comida para evitar el “por una vez no pasa nada”.

Lograr una interfaz sencilla, apta para cualquier usuario, además de una programación simplista y modularizada para facilitar el futuro desarrollo del proyecto.

2.2. Objetivos Funcionales

Los principales objetivos funcionales son:

- Crear una base de datos con la información necesaria para el correcto funcionamiento de la aplicación.
- Desarrollar una aplicación de escritorio completa que proporcione:

- Crear nuevos usuarios, modificar la información del usuario dado de alta y añadir nuevos alimentos.
 - Visualizar a través de diferentes medios el: progreso, historial y elecciones de un usuario concreto.
 - Elegir entre diferentes posibilidades recomendadas adecuadamente, o actualizar dichas opciones en caso de querer algo diferente.
 - Acceso constante al manual de usuario ante cualquier duda o inconveniente.
 - Calcular la calidad de un menú y sus diferentes características de manera automática.
- Adquirir nuevos conocimientos en el ámbito dietoterapéutico.
 - Aumentar los conocimientos en el desarrollo de aplicaciones y documentos oficiales.

2.3. Objetivos incrementales del desarrollo

El proyecto fue fragmentado en pequeños y diversos objetivos específicos para conseguir llegar al objetivo final. Se valoraron objetivos complementarios para conseguir un desarrollo incremental y mantenido, haciendo que todos los objetivos específicos encajaran de manera secuencial y facilitando así su constante desarrollo, evitando al máximo posible parches que retrasaran la creación del proyecto.

- La idea: Lo primero de todo ha desarrollar fue la idea, había que conseguir una idea original para una herramienta nueva e innovadora, algo que cubriera alguna necesidad del mundo actual. Entonces se decidió unir los conocimientos adquiridos estos años de estudio con un tema de interés actual y en auge, así surgió la idea. Entonces se hizo única, haciendo que el usuario escogiese entre diversas opciones, y teniendo en cuenta la salud de este.
- Dado que la idea ya estaba desarrollada, se paso a estructurar el proyecto teniendo en cuenta que tipo de personas usarían esta herramienta, se pensó en como poner a disposición del usuario la información que el programa iba a mostrarle. Se planificó una metodología de autoaprendizaje, e interfaz básica para evitar toda distracción y facilitar tanto el desarrollo como el uso.

- Una vez aclarado todo concepto teórico sobre el desarrollo, se pasó a pensar los objetivos a nivel de desarrollo del software, y se llevó a cabo la creación y estructura de la base de datos. Se crearon dos colecciones: los alimentos y los usuarios, con sus diferentes características para el posterior tratamiento de los datos, además de una tercera que llevase el registro de la actividad del usuario.
- Una vez pensada la mejor disposición y estructura de las bases de datos, se empezó a crear el esqueleto del programa. Se desarrolló una aplicación de línea de comandos, que de manera básica y lineal, pidiese la comprobación de usuarios y mostrara las opciones a comer.
- Una vez hecho el esqueleto, se empezaron a desarrollar los cálculos necesarios para redistribuir la información y llevar a cabo los cálculos de la manera más precisa posible.
- Ya creados los módulos y hechos los cálculos se empezó a trabajar sobre la interfaz gráfica. En este paso hubo un pequeño parche, puesto que se tuvo que adaptar el esqueleto para encajarle con las competencias de Tkinter para la interfaz gráfica.
- En este punto ya se había conseguido una aplicación sencilla y básica, y había que empezar a cubrir las necesidades del usuario. Era necesario que el usuario pudiera cambiar su elección, que tuviera a la vista la información de cada comida y la información sobre su día a día. Aquí se creó el tronco del programa, que se basa en actualizar todas las comidas cada vez que se actualiza los datos del actual día del cliente. Para esto se creó el módulo vista.
- Estando ya la aplicación funcionando y todos los algoritmos haciendo su correspondiente trabajo, tocó perfilar detalles, a simple vista de menor importancia, pero que en realidad jugaban un gran papel en el objetivo final del proyecto. Se empezó a crear el sistema de gráficos e historiales para que los usuarios pudieran aprender de sus errores, y tener un seguimiento de su avance desde el inicio del programa.
- Cuando por fin se crearon los gráficos, y el programa estaba operativo, se decidió dar la posibilidad al usuario de añadir nuevos menús a la aplicación. Así, se permite un mayor crecimiento y se acaba con uno de los grandes límites que presentaba el proyecto. Junto con este punto se creó la posibilidad de que un nuevo usuario se registrará en el programa.

Conceptos teóricos

3.1. Introducción

Se explicará todo concepto teórico interno y/o externo, usados en la metodología del programa para su correcto funcionamiento y la coherencia entre funciones, acercándose lo máximo posible a la experiencia real que se quiere conseguir con este proyecto a la hora de interconectar los datos con los métodos, dando así el mayor rigor en cuanto a veracidad.

3.2. Nutrición

La nutrición es uno de los ámbitos principales de estudio a lo largo de este proyecto, para así lograr la máxima autenticidad de los resultados. En cada cálculo y método que se realizan en el programa, se tienen en cuenta una serie de valores y datos meticulosamente estudiados para poder usarlos en el programa.

La nutrición como un avance médico, es un tema que a día de hoy esta en pleno auge. Numerosos estudios resaltan el estrecho vínculo entre un buen estilo y una buena calidad de vida en cuanto al término de salud se refiere. Ya no solo en patologías directamente relacionadas con la alimentación, sino en otro tipo de patologías más graves, en las cuales se puede reducir enormemente el riesgo de agravamiento con una buena alimentación.[5]

Reparto calórico

Aunque durante el periodo de investigación se ha demostrado que todos los nutrientes principales de un alimento son igual de buenos y necesarios, la predisposición del cuerpo ante la comida varía según la etapa del día.

Durante el desayuno somos más propensos a asimilar los nutrientes, mientras que a la hora de la cena nuestro metabolismo está más ralentizado y se debe cenar con mayor moderación, pero no es aconsejable restringirlo estrictamente.

Para ello se realizó un estudio contrastando diferentes datos y se calculó a través de los datos obtenidos el porcentaje total de cada macronutriente (hidratos, proteínas y grasas) en cada comida.

Reparto total de los macronutrientes

Hay que tener en cuenta que este proyecto ha sido diseñado para personas con diferentes patologías y que como tal, tendrán diferentes distribuciones. En un principio se pensó en todas las combinaciones posibles: baja en grasa, alta en grasa, baja en hidratos, alta en hidratos, etc.

Pero tras hablar con una doctora especialista en Endocrinología del Hospital Universitario de Burgos, María Maravi, me afirmó que:

"Nosotros no trabajamos con dietas altas o bajas en hidratos, pues son necesarios por igual para todo el mundo y por lo cual, la modificación de este parámetro solo puede traer problemas al paciente. Un diabético ha de tomar una cantidad normal de hidratos."

De este modo la especialista nos proporcionó la tabla de variaciones según el tipo de dieta (Tabla 3.1).

Reparto total calórico

Internamente se realizan múltiples cálculos, los cuales son necesarios ya sea para la correcta recomendación alimenticia del usuario, como para el correcto tratamiento de los datos.

Dietas	Hidratos	Grasa	Proteínas	Ejemplos
Normal	55 %	20 %	25 %	Cualquier persona
Baja en grasa	55 %	15 %	30 %	Problemas cardiovasculares
Alta en grasa	55 %	25 %	20 %	fibrosis quística
Baja en proteína	57 %	25 %	23 %	Insuficiencia renal
Alta en proteína	50 %	15 %	35 %	Pacientes en diálisis

Tabla 3.1: Distintos tipos de dietas y su distribución

Comidas	Grasa	Hidratos	Proteínas	carga TOTAL
Desayuno	15 %	30 %	24 %	24.75 %
Almuerzo	10 %	25 %	14 %	13.5 %
Comida	20 %	25 %	24 %	30.5 %
Merienda	20 %	10 %	14 %	11.5 %
Cena	10 %	10 %	24 %	19.75 %

Tabla 3.2: Distribución calorica a lo largo del día.

Debido a la diversidad de información encontrada en la web, fue muy complicado llegar a un punto estable donde los datos tuvieran una coherencia para todos y fueran veraces. Para que las cifras cuadrara se creó una matriz donde se dividieron las columnas por macronutrientes y las filas por el tipo de comida en concreto. Esta matriz debía tener como resultado un porcentaje que cuadrara en varios ejes y un subporcentaje que cuadrara en el total, de esta manera se tenía controlado en todo momento la veracidad de los datos. Como se puede observar en la Tabla 3.2. Se estudió para ello la diferente predisposición de las comidas y los nutrientes internos de cada uno.

Nutriscore (Algoritmo del semaforo)

De manera objetiva, se creó un método de medición para la conveniencia de los alimentos (calidad). Para que dicho cálculo mantuviese correlación con la actualidad que cubre el proyecto, se decidió usar el recurso Nutriscore, actualmente usado en España.

A continuación, se muestra en imagen el algoritmo seguido de una breve explicación:

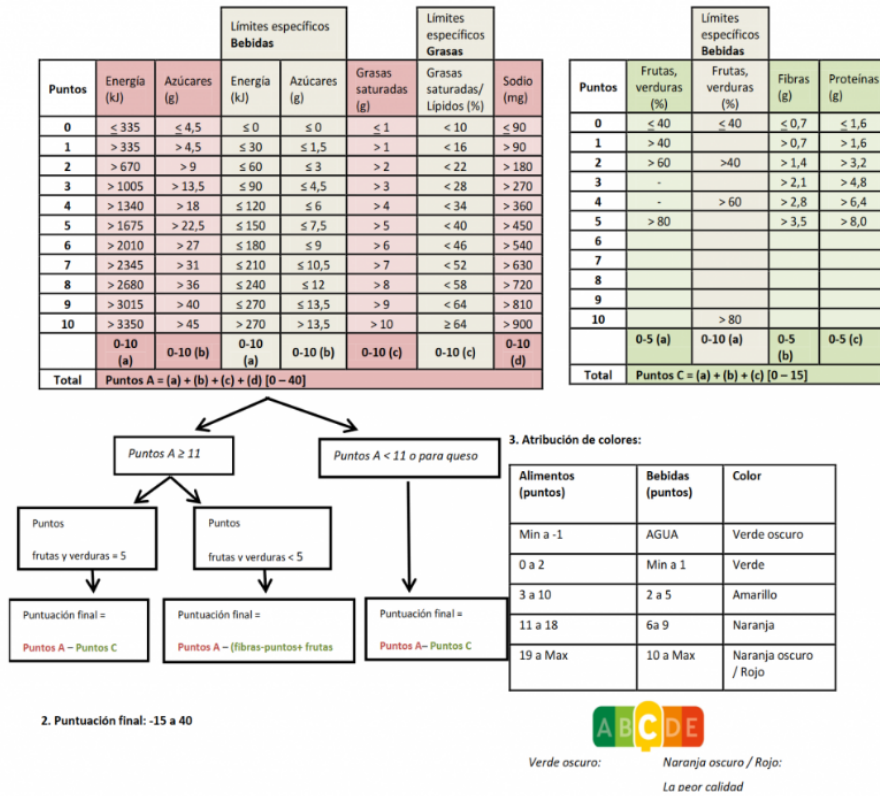


Figura 3.1: Algoritmo Nutriscore

La imagen anterior muestra el algoritmo, el cual, se basa en el sumatorio de una serie de valores negativos (Tabla roja), que son: Energía (kilojulios), Azúcares (gramos), Grasas saturadas (gramos), Sodio (miligramos). Estos valores reciben una puntuación del 0 al 10, en base a la cantidad que contenga un alimento por cada 100 gramos de comida de estas características. Y una serie de valores positivos (tabla verde) que son: frutas, fibra y proteínas. Estos valores varían entre 0 y 5, y adquieren esta puntuación de la misma forma que los anteriores.

A mayor puntuación, mayor presencia tiene sobre el resultado final, que se haya restando a la puntuación obtenida de las características negativas, la puntuación de las características positivas. En base al resultado de la diferencia se le asigna un color y letra. Como bien se ve en la tabla 3.3

Puntuación	Letra	Color
<-1	A	Verde oscuro
0 a 2	B	Verde
3 a 10	C	Amarillo
11 a 18	D	Naranja
>19	10 %	E Rojo

Tabla 3.3: Resultado final del Algoritmo Nutriscore

Imperfecciones

Este algoritmo contiene fallos, pues se basa única y exclusivamente en cifras, como 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 los cereales bajos en azúcares, bebidas edulcoradas y cosas de gran similitud nutricional, adquieren una buena calidad.

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

3.3. Prevenciones

Debido al propósito del proyecto como ayuda a la prevención o avance de enfermedades/patologías por parte del usuario, había que estudiar qué tipo de prevenciones había y donde cabía el proyecto.[8] Tipos:

- **Prevención Primaria:** Evita la adquisición de la enfermedad. Estas técnicas actúan suprimiendo los factores desfavorables antes de que puedan llegar a generar la enfermedad.
- **Prevención Secundaria:** Detectar la enfermedad y evitar su posible progresión. Interviene cuando se inicia la enfermedad, y su función principal es que el diagnóstico y tratamiento precoz mejoren la evolución y control de las enfermedades.
- **Prevención Terciaria:** Comprende aquellas medidas dirigidas al tratamiento y a la rehabilitación, además de la ralentización de su progresión. Interviene cuando las lesiones son irreversibles, y la enfermedad está arraigada, pudiendo llegar a ser crónica.

Análisis:

Respecto a la **prevención primera**, una buena alimentación junto con una serie de hábitos de vida saludables puede impedir enormemente el desarrollo de la enfermedad, siendo la recomendación de primera elección por los profesionales a sus pacientes.

En la **prevención secundaria** también entra esta categoría. En patologías como la diabetes o problemas cardiovasculares, la dieta es vital para su tratamiento. Un buen hábito de vida y de alimentación puede tener incluso mejores resultados que la medicación en algunos casos (como se observa en el estudio de prevención de diabetes [16]).

En el caso de la **prevención terciaria** es más complicado de asegurar. A falta de estudios concluyentes, no se puede afirmar con exactitud que el proyecto cubra este tipo de prevención en estadios tan avanzados de la enfermedad.

Lo que si es cierto es que una buena alimentación, junto con actividad física moderada a lo largo de la vida, siempre es recomendable para la salud de cualquier persona.

3.4. Enseñanza

La idea principal del proyecto es evitar otra aplicación repetitiva sobre la gestión o impartición de dietas, las cuales suelen ser abandonadas por la mayoría de los españoles al paso del tiempo.

Para ello se estudió varios métodos de enseñanza, aprendizaje a través de métodos multimedia, etc.

Métodos multimedia

En los últimos años, el desarrollo de aplicaciones tanto móviles como de escrito han sido integradas activamente en los diferentes procedimientos de enseñanza. Se considera que las aplicaciones a día de hoy ofrecen muchas ventajas al mundo de la enseñanza:

- Comunicación a tiempo real entre el método de enseñanza y el alumno, estando siempre al alcance de éste.

- Cómoda distribución de tareas, siendo más fácil el orden y la organización.
- Ayuda a la superación de las barreras geográficas.

En educación, se ha pronosticado a que lo largo de los años dominará el uso de la Tablet. Por ello, parte de la progresión del proyecto es su extensión a Tablets y Smartphones. [4]

Autoaprendizaje

Debido a que el uso de este tipo de aplicaciones carece de una figura presenta durante la fase de aprendizaje del usuario, se buscó un método en el que el usuario por su propia índole, aprendiese y poco a poco fuese adecuando su nuevo habito de vida, en base a sus propias elecciones. Esto otorga al usuario versatilidad, dándole la opción en todo momento de elegir que es lo que mas le apeteciese, recomendando la mejor opción y complicando que el usuario haga una mala elección. De esta manera se juega con la idea de hacer creer al usuario que es él el que toma las buenas o malas decisiones.

3.5. Autoaprendizaje del valor TMB

Pese a que en ningún momento se llevo a implantar, se desarrolló un estudio sobre inteligencia artificial o redes neuronales en la aplicación para hacer del programa un sistema único. Llegándose a crear una neurona artificial con la intención de programar una función que fuera la *autoselección* de los menús, pero esto solo reforzaría las elecciones más comunes del usuario que no tienen por qué ser adecuadas.

Finalmente, se ideó un sistema de autoaprendizaje basado en el reajuste del valor de una variable, la cual cambia en base a la progresión del usuario. Esto permite tener un sistema único que resuelve el principal problema del calculo del TMB, la generalidad.

3.6. Métodos de recomendación

Se meditaron diferentes métodos de recomendación:

- Filtro colaborativo basado en modelos: Utilizan los datos para ajustar modelos que después pueden ser utilizados para proponer recomendaciones.
- Filtro colaborativo basado en usuario: Es un tipo de filtro basado en memoria, y se basa en recomendar lo que usuarios similares a nuestro usuario han escogido.
- “Vecino más cercano”: Su cálculo se basa en la correlación de Pearson, y es muy similar (por no decir idéntico) al filtro colaborativo basado en usuarios.

Todos estos métodos son ampliamente utilizados por grandes distribuidores como Amazon, Netflix ...

Pero para este proyecto presentaban dos grandes inconvenientes:

- Como bien se vió en la asignatura de gestión de la información, son métodos que requieren de muchos usuarios y productos para que funcionen de manera precisa, sino las recomendaciones pueden ser totalmente aleatorias, dando paso a una muy mala recomendación.
- La más importante, si este proyecto es creado por el desconocimiento extendido sobre la buena alimentación y sus beneficios, no se puede comparar a un usuario con el resto de ellos, pues a la larga el programa se “estropeará”, y pasará a dividir a las personas en dos grandes grupos: los que se cuidan bien y los que no. Y es precisamente lo que se quiere evitar.

Por ello, pese a ser uno de los filtros menos recomendados en la asignatura de Gestión de la Información, para este proyecto fue el más apropiado. Se pasó a usar un filtro basado en contenidos, por lo general este filtro necesita saber las características del usuario y del contenido de los productos (en este caso alimentos), para poder realizar la recomendación. Esto obviamente tiene un gran coste de información y por ello no es recomendable. Pero en este proyecto se parte con esa información. Básicamente, sabiendo exactamente las características que el usuario busca en esa comida, se le recomienda dicho producto (Menú) que más se adapte a lo que busca.

Técnicas y herramientas

Apartado cuyo objetivo es presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto. Si se han estudiado diferentes alternativas de metodologías, herramientas, bibliotecas, etcétera.

Se comenta los aspectos más destacados de cada opción, con un repaso somero a los fundamentos esenciales y referencias bibliográficas para que el lector pueda ampliar su conocimiento sobre el tema.

4.1. ¿Por qué Python?

La razón es la sencillez y capacidad de este lenguaje para el análisis y tratamiento de los datos, gracias a librerías como Pandas o Numpy. Base gran parte de la metodología de análisis usada en este proyecto en el libro: Python for Data Analysis. [13]

Las distintas posibilidades cotejadas antes de empezar fueron: Java, Android, R y Python. El primero en ser descartado fue Java, tras un estudio inicial sobre como se quería desarrollar la aplicación. Después de buscar entre diferentes fuentes; Java no era la mejor opción, dichas fuentes siempre te orientaban hacia Python y/o R, debido a la cantidad de librerías enfocadas al análisis y tratamiento de los datos que estos poseen.

La razón por la que se planteo Android fue por que este trabajo está desarrollado para unir el mundo de la dietoterapia, y las ciencias de la Salud con la informática. La forma más clara y rápida de llevar al usuario dichas tecnologías es a través de un SmartPhone, pero se acabó descartando debido a la falta de conocimientos sobre sistemas Android. A estas alturas ya solo

existían dos opciones: Python y R.

Tras indagar superficialmente sobre ambos lenguajes para el análisis y el tratamiento de datos, se llegó a la conclusión que ambos lenguajes tienen una forma de trabajar muy similares, y debido a que durante los años de estudio el alumno ha trabajado en numerosas ocasiones con Python, se decantó por Python.

4.2. Metodología

Introducción

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

Excell y Pandas

Se ha usado la herramienta de Microsoft Excel, para simular una base de datos, trabajando los datos en formato DataFrame. Los DataFrame son proporcionados por la librería Pandas. Existen diferentes hojas de cálculo, que funcionan como colecciones de una base de datos. Estas hojas son explicadas más adelante.

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. Los alimentos se tratan en el momento en el que se registra el usuario, de manera que se separan las comidas de la lista principal, creando 5 listas (Desayuno, Merienda, Comida, Almuerzo y Cena), se tratan por separado y se filtran. Cada modificación o inserción de un dato o usuario se hace sobre el DataFrame, el cual sustituirá a la base de datos durante el guardado.

4.3. Técnicas

Introducción

En este apartado se hablará de manera breve de las técnicas usadas durante el proyecto, y la razón por la que se escogieron dichas técnicas.

Posiblemente haya argumentos repetidos en otros apartados similares, por lo que se expresará exclusivamente finalidad y razonamiento.

Aprender a aprender

“Aprender a aprender supone disponer de habilidades para iniciarse en el aprendizaje y ser capaz de continuar aprendiendo de manera cada vez más eficaz y autónoma de acuerdo a los propios objetivos y necesidades.

Esta competencia tiene dos dimensiones fundamentales. Por un lado, la adquisición de la conciencia de las propias capacidades (intelectuales, emocionales, físicas), del proceso y las estrategias necesarias para desarrollarlas, así como de lo que se puede hacer por uno mismo y de lo que se puede hacer con ayuda de otras personas o recursos. Por otro lado, disponer de un sentimiento de competencia personal que redunde en la motivación, la confianza en uno mismo y el gusto por aprender. Significa ser consciente de lo que se sabe y de lo que es necesario aprender”. [7]

Es decir, el autoaprendizaje no solo ayuda al usuario a aprender de manera autónoma, sino que le ayuda a que todo aquello que ha conseguido, para el sea, de alguna forma, una meta, algo que lograr, algo suyo, que no se le va a olvidar tan fácilmente, como información memorizada, sino que será el resultado de pequeños logros personales que integrarán al usuario, una nueva información, como si fuese suya.

Tasa Metabólica Basal - TMB

La tasa metabólica basal (TMB) es el gasto calórico de una persona a lo largo del día. A este cálculo se le multiplica un valor relacionado con la actividad física del usuario y como resultado tenemos las kilocalorías diarias que el usuario gasta al día o lo que es equivalente, que el usuario debe tomar para mantenerse en su peso.

El cálculo es genérico, por esta razón es un método cada vez más en desuso. Aun así, sigue siendo un de los métodos más utilizados para la recomendación dietética por especialistas en la materia.

La fórmula del TMB es la siguiente:

Ejercicio	Valor de ActividadFisica
Poco ejercicio	1,2
Ejercicio ligero(1-3 dias/semana)	1,35
Ejercicio Moderado (3-5 días/semana)	1,55
Ejercicio fuerte (6-7 dias/semana)	1,725
Ejercicio muy fuerte (dos veces al día)	1,9

Tabla 4.4: Valor de la actividad física en TMB

Hombres:

$$TMB = ((10*Peso(kg))+(6,25*Altura(cm))-(5*edad)+5)*ActividadFisica \quad (4.1)$$

Mujeres:

$$TMB = ((10*Peso(kg))+(6,25*Altura(cm))-(5*edad)-161)*ActividadFisica \quad (4.2)$$

Donde actividad física se corresponde con los siguientes valores mostrados en la Tabla 4.4. [10] Para cumplir el objetivo de la persona (subir o bajar de peso), se suele añadir o restar un valor estandarizado. Este valor se encuentra entre 300 y 500 calorías si se quiere adelgazar o subir aproximadamente un kilogramo por semana [15].

Es este proyecto para evitar la generalización de la fórmula, lo cual es la principal razón de que la fórmula este siendo sustituida por la toma de medidas corporales, se crea una variable que inicialmente empieza en 500 calorías, pero que varía en base al progreso del peso del usuario. Si el usuario se pasa o se queda corto del objetivo normal (con un rango de error del 15 %), este valor variará ajustándose cada vez más al usuario (si el usuario se excede se le reducirán 50 calorías, si se queda corto se le sumarán 40).

Tratamiento de Excepciones

A través del tratamiento de excepciones, y con el uso de la librería **messagebox** se informa al usuario de las distintas situaciones que se pueden dar durante el uso del programa. Existe tres tipos principales de cuadros de textos, que emergen informando al usuario de algún tipo de problema o simplemente de que una acción ha finalizado correctamente.

Información

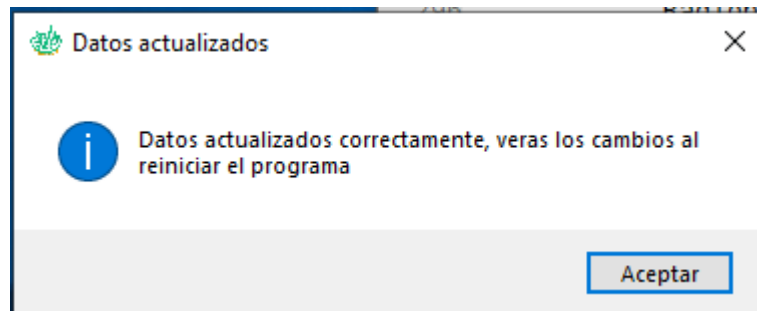


Figura 4.2: Ventana emergente de información

Muestra la información de que un proceso ha concluido correctamente. También explican si el usuario ha de realizar algo una vez terminado el proceso para que termine la completa finalización de este.

Aviso

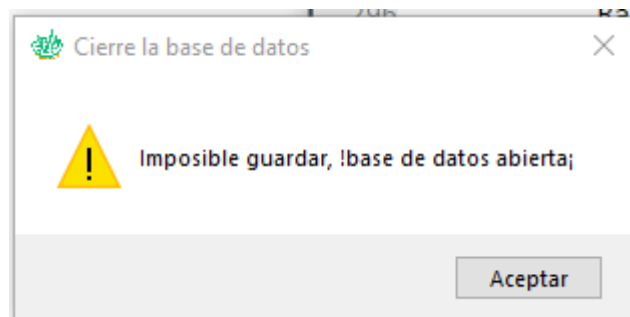


Figura 4.3: Ventana emergente que avisa que una acción no se ha podido realizar.

Pantalla que advierte que algo no ha salido bien. El usuario puede seguir con el uso del programa, pero si quiere realizar la acción que estaba realizando, deberá seguir las instrucciones proporcionadas en el cuadro de aviso.

Error

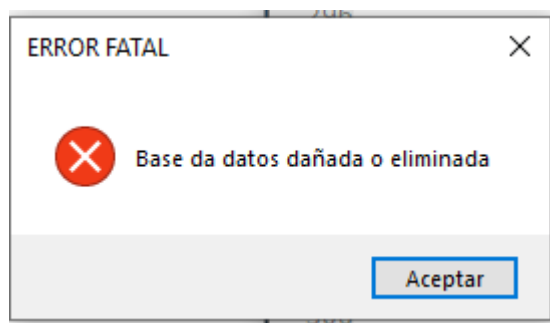


Figura 4.4: Ventana emergente que muestra que ha ocurrido un Error.

Pantalla que advierte de un error grave, en el que se ha corrompido la estructura de datos del programa.

A través de estos cuadros se traduce los diferentes problemas que puedan surgir en la utilización del programa; de manera gráfica y entendible por el usuario, haciendo el funcionamiento de este lo más transparente al usuario posible.

Diseño

Diseño simple, se evita crear extensos árboles de ventanas y frames, creando una navegabilidad sencilla e intuitiva. Debido al principal objetivo del proyecto, esta aplicación está orientada a todos los públicos, por lo que se procura que la interfaz sea simplista, y fácil de entender.

Se basa en un menú principal ramificado en tres vertientes: Usuario, Dieta y Registro (en el programa aparecen con otros nombres). En la vertiente usuario, esta la información del usuario y la posibilidad de cambiar dichos datos para un avance del programa. En la rama de la dieta se encuentra el tronco principal de la aplicación, se muestra las recomendaciones alimenticias, a parte de darte libertad a la hora de escoger y decidir que deseas comer en el día de hoy. Por último estaría el registro o historial, es decir, todo aquello que necesites para llevar un registro de tu progreso y concienciar de esta manera al usuario.

Respecto a los colores, se decido un tema básico que no agote la vista del usuario, ni tenga múltiples colores deslumbrantes, da la posibilidad al usuario de elegir entre una serie de estilos predefinidos para que escoja el que mejor se ajuste a sus gustos y necesidades.

4.4. Estructura del programa

Introducción

La estructura es una variación del modelo-vista-controlador, separando el módulo HealthApp (que funciona de Main), del resto de módulos, que son los que implementan la estructura CAV (Cálculos, Administración y Vista).

1. **HealthApp** – Tronco del programa
2. **Vistas** – Relacionado con la parte visual
3. **CalculosDieta** – Todo cálculo usado en el programa
4. **AdminBase** – Todo proceso relacionado con la base de datos

Cálculos, Administración y vista

Podríamos llamar propiamente a este modelo el CAV, y se basa en separar toda función que se vaya a crear en el programa en estos tres pilares. Cada vez que la lógica de una función necesitaba un hueco donde encajar y ninguno de los módulos creados lo sostenía, se iba creando un nuevo módulo. Debido a la forma de trabajar del alumno, la estructura acabó cogiendo esta forma. Por un lado, todo elemento relacionado con la base de datos, por otro lado todo cálculo matemático realizado en el programa y por último toda función que intervenga en la interacción usuario-aplicación.

Para crear una estructura, y no tener un código modular sin coherencia, se llevó a cabo una serie de reglas sencillas:

- Si su finalidad es puramente visual – Vista
- Si realiza cálculos u operaciones matemáticas, indistintamente de si sea sobre la dieta -CalculosDieta
- Si interacciona y guarda datos en la base de datos -AdminBase

- Si mezcla alguna de estas funcionalidades, por ejemplo, guardar en la base de datos y mostrar por pantalla algo, se tendrá en cuenta cual es su principal finalidad, por ejemplo, se guardan estilos que se cargaran automáticamente para el diseño del programa, obviamente su finalidad es puramente visual, y guardar en la base de datos una transacción necesaria.
- Cualquier objeto, ventana, tabulación, etc. Que siempre vaya estar ahí al HealthApp, pero sus modificaciones se repartirán entre los diferentes módulos.

Estas reglas, a día de hoy, han cumplido con todas las funciones que se han creado en el programa dando el resultado previsto.

4.5. Librerías

Pandas

La librería desarrollada por Wes McKinney, Pandas, es una librería usada para el tratamiento de los datos como estructuras. Debido a que este proyecto se enfoca en el análisis de datos, y no en el tratamiento de bases de datos, se decidió crear 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”. De esta manera se cargaba los datos, pero en listas desestructuradas que complicaban el posterior análisis de la información.

Acto seguido basándose en el libro: Python for Data Analysis [13], se decidió probar con Pandas, esta librería permitía leer los datos de los archivos “.xlsx”, en forma de DataFrames. Simplificando el análisis de los datos, permitiendo tratarles tanto como DataFrames como en forma de vectores.

Numpy

Como se ha nombrado anteriormente, el libro: Python for Data Analysis [13]; Hace especial énfasis en la conveniencia del uso de Numpy para el tratamiento y análisis de los datos. Es la principal librería sobre la que se ha trabajado a lo largo de la carrera en cuanto a análisis de datos en Python, y una de las mas recomendadas por varias fuentes de información. La librería Numpy es la encargada, del tratamiento, procesamiento y cálculo de todos los datos que internamente realiza el programa.

Interfaz Gráfica

La librería utilizada para realizar y diseñar la interfaz gráfica fue Tkinter, tras indagar en diferentes fuentes, se decantó por Tkinter debido al desconocimiento sobre el funcionamiento de interfaces gráficas con Python. Entre las opciones que se barajaron se encontraban: Tkinter, WxPython, PyQt y PyGTK.

Tkinter presentaba una serie de ventajas: viene preinstalada con Python, es fácil de aprender y además cuenta con una documentación amplia y extensa. Pero entre sus desventajas, se encuentran: la escasez de recursos gráficos, la sencillez implica poca variedad de elementos y funciones (complicando en numerosas ocasiones la navegabilidad) y la lentitud que este tiene debido a que dibuja cada elemento sobre cada pantalla, en tiempo de ejecución.

La segunda opción que se planteo, fue WxPython. Presentaba grandes ventajas como: la rapidez, la flexibilidad que este ofrecía y la variedad opciones que tiene para crear una interfaz gráfica compleja. Después de ser meditado se llego a la conclusión de que todo lo que ofrecía WxPython, era innecesario para la interfaz tan simplista que necesitaba el proyecto, y el aprendizaje era más complejo, además de tener recursos limitados, y tener menor cantidad de documentación al alcance del alumno. Su mayor inconveniente es que tiene una comunidad muy activa, la cual está constantemente insertando cambios y creando problemas de compatibilidad.

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

Matplotlib

Librería de Python extendida en su uso, para el análisis y representación de los datos. Se utiliza a lo largo del proyecto para crear diferentes gráficos e histogramas para dar una perspectiva más visual de los resultados al usuario.

Pillow

Usada en su versión 6.0. Es una librería que permite la carga y posterior muestra de imágenes en una aplicación Python. Es usada en numerosas

ocasiones a lo largo del proyecto, para visualizar el logotipo, o diferentes imágenes guía que ayudan al mejor entendimiento del programa.

webbrowser

Módulo que proviene de una interfaz de alto nivel para mostrar archivos web a los usuarios.

os

Librería utilizada durante el proyecto para la carga del manual en PDF por parte del programa. Es una librería cómoda que permite acceder y mostrar varios elementos que se encuentran en el sistema.

datetime

Librería normalmente preinstalada con Python, que da un control sobre la fecha y la hora del sistema. Sumamente útil para el control de registros e historial.

auto-py-to-exe

No es usada durante el desarrollo del proyecto, pero si al finalizar dicho desarrollo para crear un archivo ejecutable (.exe). Utiliza internamente la librería pyinstaller, y hace uso de la librería Tkinter para otorgar una interfaz gráfica al comando pyinstaller, haciendo más fácil su uso.

4.6. Módulos

Cálculos Dieta

Módulo encargado de los cálculos del programa. En el se encuentra el motor del sistema de recomendación, encargándose de:

- Calcular la calidad de los alimentos en base a sus características.(Algoritmo Nutriscore)
- Calculo y reajuste de la Tasa Metabólica Basal (TMB)
- Reparto de kilocalorías y los diferentes macronutrientes, teniendo en cuenta la comida del día y las necesidades que este aplica.

Comida	Porcentaje(%)	Descripción
Desayuno	24,75	Alta carga calórica, rica en hidratos
Almuerzo	13,5	Baja carga calórica.
Comida	30,5	Mayor carga calórica, eje central.
Merienda	11,5	comida de paso y casi prescindible.
Cena	19,75	Alta carga calórica pero con moderación.

Tabla 4.5: Porcentajes totales de cada comida

- Aplicación de la formula del sistema de recomendación

Reparto Calórico entre comidas.

Como se aprecia en la Tabla 4.5, se tiene en cuenta no solo la carga calórica en cada comida, sino la distribución de este porcentaje entre los distintos nutrientes. (Consultar Tabla 3.2)

Se calcula el número de Kilocalorías que se tiene que tomar de hidratos, grasas y proteínas, en base al tipo de dieta recomendable para su patología. **distribuciónDeMacronutrientes** toma como parámetros las kilocalorías diarias calculadas previamente por la función `calculoTMB`, y el tipo de dieta del usuario en base a su patología, y te devuelve una lista de los macronutrientes diarios, repartidos en gramos y las kilocalorías totales en el siguiente orden: `ListMacDiarios [Kilocalorías, Hidratos, Proteínas, Grasas]`.

Formula del Sistema de recomendación

Motor del sistema de recomendación. A través de la formula:

$$DiferenciaMacronutriente = \frac{(KMCD - A)}{((KCDT - KMM) + (KMTC - KTM))} \quad (4.3)$$

Donde:

- **KMCD:** Número de Kicalorías del macronutriente (Hidratos, proteínas o grasas) de la Comida (Desayuno, almuerzo...) que debería llevar en esa comida
- **A:** Actual número de kicalorías de ese macronutriente que llevo en todo el día (Si estoy en Comida: desayuno+almuerzo+comida).

- **KCDT:** Kilocalorías que debería comer en esta comida concreta para este macronutriente concreto.
- **KMM:** Kilocalorías del macronutriente concreto del menú.
- **KMTC:** Kilocalorías totales que debería comer en esta comida.
- **KTM:** Kilocalorías totales que tiene el menú a recomendar.

La fórmula es el eje central del filtro de recomendación basado en contenidos. Se evalúan diferentes características de los tres aspectos claves a la hora de la recomendación según este proyecto: Usuario, Alimento, y el momento del día.

De esta manera cuanto mayor sea la diferencia entre lo que debo llevar y lo que llevo comido, y menor sea la diferencia entre lo que deseo y las características del alimento, mayor será la puntuación de ese alimento para su recomendación (**Nota: Se ordena de mayor diferencia a menor**). Por último, se realiza la media entre los tres macronutrientes principales:

$$DiferenciaTotal = \frac{(Diff.Hidratos + Diff.Proteina + Diff.Grasas)}{3} \quad (4.4)$$

En el caso de que algún macronutriente dé negativo, no se tendrá en cuenta.

AdminBase y estructura de la base de datos

En este apartado se expone la estructura la base de datos y de como se cargan, guardan y trabajan los datos desde el módulo **AdminBase**, el cual, es el encargado del manejo de los datos, y la traducción con el sistema de archivos.

CargaBaseDeDatos y guardaDatos

Estas dos funciones cargan y guardan los datos de/en la base de datos respectivamente. Nada más iniciar la ejecución del programa, almacena los datos leídos de la base de datos, en una serie de DataFrames independientes. Se trabaja con cada uno de ellos sin tener en cuenta al resto y se guarda el resultado de los cambios producidos durante la ejecución del programa, sustituyendo la base de datos por los DataFrames.

Estructura de la base de datos

La base de datos es almacenada en diversas hojas de cálculo, simulando de esta manera diferentes colecciones de una base de datos. Estas colecciones serían:

ALIMENTOS

La hoja de alimentos, es una hoja de menús ya construidos. Esta hoja consta de doce campos:

La clave primaria de esta colección es el nombre del menú. Los campos son:

Nombre	Calorías	Grasa	Saturadas	Hidratos	Fibra	Azúcares	Proteína	Sodio	Tipo	LRE	Calidad
Tarta de queso	547	18,9	9,4	75,9	1	41,6	17,5	0,7	10	0	4
Torrijas	566	39,6	8,4	38,9	1	12	12,4	0,7	26	0	4
Flan	244	7,4	3,1	37,4	1	33,3	7	0,7	26	0	4

Figura 4.5: Estructura base de datos alimentos

Nombre (Único), Calorías, Grasa, Saturadas, Hidratos, Fibra, Azúcares, Proteínas, Sodio, Tipo, LRE (Last Recently Eat) y Calidad. Los primeros diez campos son las diferentes características del menú, mientras que los otros tres son parte de la gestión de dichos menús.

El tipo marca el tipo de comida que es, un mismo menú puede ser a la vez desayuno, almuerzo y merienda, cena y comida, etcétera. Por ello se planteo dos posibilidades.

1. Cadena de caracteres, con la combinación de la inicial o nombre de las diferentes combinaciones de tipos de comida y la posterior separación de estas. Esto dio diversos problemas con la carga y filtrado de los datos.
2. Cadena de bits: desayuno-almuerzo-comida-merienda-cena, donde el valor 1 sería si es valido para esa comida y 0 si no lo es.

Se decidió escoger la segunda opción. Una vez creado los métodos que traducen el sistema numérico en una cadena de bits, el tratamiento en conjunto de las combinaciones como un simple número simplificó las cosas. Ejemplo: 1-1-0-1-0

Significa que es desayuno, almuerzo y merienda, pero no comida o cena, luego esto se traduce en el número decimal, que en este caso sería 26.

El LRE lleva una métrica de la frecuencia con la que el usuario toma esa opción, literalmente significa “last recently eat” (El más reciente ingerido),

que en este caso hará de contador. En la expansión del proyecto (Explicado en el apartado 7), este dato sería el único encontrado de manera Local, ayudando a personalizar aún más la aplicación para una mejor experiencia de usuario.

La Calidad, como el propio nombre indica, es la calidad del alimento, es la variable que marca la diferencia entre 100 kilocalorías de ensalada y 100 kilocalorías de azúcar. Tiene un rango de valores de la A a la E, siendo A una comida saludable, y E una comida poco recomendable. Internamente el programa lo trata de forma numérica. Este valor se halla a través del algoritmo Nutriscore.

Usuarios

La hoja de usuarios alberga los datos de todos los usuarios que usan la aplicación. Está estructurada de forma que contenga los datos principales del usuario, así como, los datos necesarios para el sistema de recomendación.

id	nombre	apellido	password	sexo	edad	altura	peso	actividad	patologia	tipo
71308125	jesus	carro	1234	H	24	178	73,4	4	0	subir
1234	Guille	Calvo	1234	H	21	177	69	3	0	subir
1	Pedro	Garcia	1234	H	21	180	76	1	1	mantener
3	Ana	Perez	3	M	21	168	55	2	1	mantener

Figura 4.6: Estructura base de datos Usuarios

La clave primaria sería el DNI del usuario, el cual, se encuentra dentro del campo ID. La hoja se estructura de la siguiente manera: Id (PK)-nombre-apellido-password-sexo-edad-altura-peso-actividad-patología (FK de la tabla Patologías) y tipo.

Tanto **id**, como **nombre**, **apellido** y **password**, son los datos privados del usuario, los cuales, son simplemente informativos y no tienen ningún valor adicional en el cálculo de la dieta y los resultados. En cambio, **el sexo**, **la edad**, **altura**, **peso**, **actividad**, **patología** y **tipo**, **influyen de manera directa con el cálculo de la dieta**. A tener en cuenta, que el valor del campo patología es numérico para mayor privacidad y conexión con la base de datos de Patologías, pues es el id de cada patología. En el caso de valer -1, significa que el usuario no tiene ningún tipo de patología y su uso es exclusivamente para el aprendizaje y seguimiento de la dietoterapia adecuada. La actividad tiene valores entre 0 y 4, donde cero significa el máximo nivel de sedentarismo y 4 el máximo nivel de actividad. Por último el tipo, hace referencia, a los objetivos que el usuario tiene respecto a si

mismo, si quiere mantenerse, subir de peso, o bajar.

Patologías

En esta hoja, la cual se encuentra dentro de las BaseDeDatosDeAlimentos, está la información básica de las patologías: **nombre**, **id**, y el **tipo** de dieta que lleva alto o bajo en carbohidratos, proteínas o grasas. Esto se carga al inicio del programa, se comprueba si el usuario padece alguna patología y se selecciona el tipo de dieta correspondiente.

id	nombre	dieta
0	sin patologia	normal
1	Colesterol	baja en grasa
2	Insuficiencia Renal	baja en proteina
3	Pacientes en dialisis	alta en proteina
4	fibrosis quistica	alta en grasa

Figura 4.7: Estructura base de datos Patologías

Registro

Base de datos que se encuentra en la ruta: assets/RegistroMejora.xlsx, y se encarga de llevar un registro de la mejora del usuario en la aplicación. Esta base de datos guarda los datos imprescindibles para el ajuste de la formula del TMB respecto al usuario concreto.

En el momento que un usuario se registra, tiene tres posibilidades de tipo de dieta:

1. **bajar:** Se crea un valor inicial igual a -500
2. **Mantener:** Se crea un valor inicial igual a 0
3. **Subir:** Se crea un valor inicial igual a 500.

El valor inicial a la hora del registro es el suplemento que se añade al resultado de la formula del metabolismo basal, para que las calorías a ingerir por el usuario sean las necesarias para lograr el objetivo que se ha propuesto a través del tipo de dieta. Esta base de datos se estructura de la siguiente manera: Esto se debe a:

Fecha	dni	peso	valor	tipo
2019-05-16	71308125	73,4	500	subir
2019-05-16	1234	69	500	subir

Figura 4.8: Estructura base de datos Registro

1. **Fecha:** Guarda la última vez que se editó la información. De esta manera se puede calcular el peso que se debería haber progresado hasta dicha fecha. No admite cambios de menos de una semana, pues el resultado podría alterar el ajuste del valor.
2. **dni:** DNI del usuario para saber de quién es el registro que se está llevando.
3. **Peso:** El peso de la última vez que el usuario edito la información respectiva a su peso. Esto ayuda al seguimiento de la dieta y sus objetivos.
4. **Valor:** Valor que más tarde se suma al cálculo TMB y que ayuda a cumplimentar mejor su objetivo.
5. **tipo:** Tipo de dieta u objetivo de la dieta del usuario. Sirve para saber si el usuario a cambiado de objetivo y en caso de no hacerlo, ver cómo va el progreso del usuario.

Vista

En el módulo vista se almacena toda función que tendrá influencia en lo que el usuario ve, es decir:

1. Actualización de pantallas,
2. Muestra de datos,
3. Transacciones entre ventanas,
4. Actualización de los gráficos,
5. Etcétera.

Reconstrucción de frames

Cada vez que se selecciona una opción, el resto de las opciones cambian aplicándose el sistema de recomendación programado. Se van ajustando las opciones a las necesidades del usuario. Tkinter no dispone de ningún método eficaz que permitiese actualizar un Frame. Por ello, cada vez que se selecciona una opción, se crea un nuevo frame que sustituye al actual, llamándose de manera recursiva con distintas opciones sobre el sistema de recomendación.

Para evitar errores de coherencia el programa solo actualizará aquellos Frames no bloqueados por el método seleccionar, sacrificando un pequeño porcentaje de la precisión, para conseguir un funcionamiento más fluido sin errores de compatibilidad o múltiples selecciones.

HealthApp

Es la columna vertebral del programa, el cual, se encarga de cargar y procesar toda la información relevante que más adelante se va a ir editando en el programa. HealthApp tiene una única función y el resto se divide en clases que se instancian en esta función.

Hace de flujo de entrada y salida, donde una condición simula un interruptor permitiendo que la aplicación original, distribuida en clases, se lance o se cierre.

Todo se basa en una jerarquía de objetos, en el que el objeto principal (Menú principal), contiene otros tres objetos que estos a su vez contienen otros, creando una simple estructura de árbol, entendible por cualquier programador.

Se crea una clase por tipo de comida barajada. Por lo que cada vez que se crea de manera recursiva el mismo Frame, se está instanciando la clase contenida.

4.7. Herramientas

Nos centraremos en mencionar de manera breve, las herramientas que se han usado para el desarrollo del programa.

Anaconda

Distribución de código abierto, la cual, está indicada para el análisis, procesamiento y computo de datos, de Python y R, trae consigo una serie de programas y características entre los que destacaremos: Spyder y VisualCode.

Se decidió usar anaconda, debido a su principal función, el análisis, procesamiento y computo de los datos.

Spyder

Spyder es un entorno de desarrollo incluido en el paquete de anaconda.

Antes de empezar se probaron varios entornos como: NoteBook, Py-Charm, VisualCode y eclipse (añadiendo el API de Python), pero se acabó decantando por Spyder.

GitHub

Proyección de la metodología GIT para el almacenamiento y control de versiones, sin duda GIT es actualmente la estructura más usada en el mundo de la programación, y GitHub, una de las herramientas más conocidas y valoradas que hay.

GitKraken

Potente interfaz gráfica multiplataforma desarrollada por Electron, que nos permite controlar de forma sencilla y cómoda nuestro repositorio GitHub, permitiendo que el uso y control del GitHub fuera más ameno y rápido, sin duda es una de las mejores aplicaciones que he probado, y de las que más me han ayudado a lo largo del proyecto.

MakeText

Herramienta de escritorio, utilizada para la preparación de documentos en LaTeX. Fue utilizada durante el desarrollo del proyecto para plasmar las memorias y anexos, que formarían la documentación del proyecto.

Aspectos relevantes del desarrollo del proyecto

5.1. La idea

Con el objetivo de innovar se crea esta idea propia, buscando la originalidad del proyecto y renovación de los métodos tradicionales. Para ello, se lleva a cabo una investigación sobre las necesidades que sufren las personas actualmente, con las ultimas corrientes de moda y que además pudiera resultar útil. La salud y la buena calidad de vida son temas de vital importancia para las personas, que con el paso del tiempo cobran cada vez más protagonismo.

Una vez establecida las bases del proyecto se analizan los posibles métodos u opciones para conseguir el fin del proyecto:

- Deporte: Se excluye por la inmensa cantidad de aplicaciones de planificación de entrenamiento que existen a día de hoy.
- Nutrición: Inicialmente se descartó la idea debido a las mismas razones que lo anterior. Existen multitud de aplicaciones relacionadas con la planificación alimenticia.
- Educación nutricional: Enfoque de la nutrición, visto desde el punto de vista del auto-aprendizaje, para usuarios sin experiencia. Se optó por este desarrollo, debido a la falta de plataformas o aplicaciones, cuya finalidad sea inculcar a los usuarios este nuevo habito.

Finalmente, la idea consiste en un sistema de recomendación nutricional, con la finalidad del aprendizaje de un hábito de vida saludable.

5.2. Formato de trabajo

Se desarrollo una metodología ágil de forma incremental. Se empezó formalizando las bases hasta llegar a la etapa final. No se pasó 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 detallada.

De esta forma, se fueron creando todas las características progresivamente. Se priorizó el desarrollo del proyecto frente a la velocidad de dicho desarrollo. Debido al sistema de organización por etapas, se consigue un progreso fluido. Utilizando los recursos substraídos de la etapa anterior, impidiendo parches en el código o incoherencias.

Se desarrollo en estas tres etapas para el perfeccionamiento del código. Se desarrolla primero una aplicación por interfaz de comandos, permitió perfeccionar la aplicación, sin tener que preocuparse por otro tipo de problemas. Se crea una estructura sólida, que posteriormente heredaría la fase de la interfaz gráfica.

5.3. Algoritmos del programa

Se desarrollan desde cero una serie de algoritmos o sistemas de tratamiento de los datos. Haciendo de esta parte algo propio de este proyecto.

Algoritmos de recomendación

Es el algoritmo que estructura todo el sistema de recomendación, 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, y el usuario nunca aprendería a llevar un estilo de vida saludable; además de que podría haber una desigualdad inmensa entre las calorías ingeridas y las necesitadas.

Se pensó en hacer una formula basada en la diferencia calórica, cosa que se adecuaba más a las necesidades del usuario, pero creaba un desajuste en los macronutrientes, rompiendo la idea de dieta equilibrada. Por ello, se ideo otro prototipo de formula.

La fórmula 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 específico y lo que el alimento tiene.

$$DiferenciaMacronutriente = \frac{(KMCD - A)}{((KCDT - KMM) + (KMTC - KTM))} \quad (5.5)$$

Donde:

- **KMCD:** Número de Kcalorias del macronutriente (Hidratos, proteínas o grasas) de la Comida (Desayuno, almuerzo...) que debería llevar en esa comida
- **A:** Actual número de kcalorias de ese macronutriente que llevo en todo el día (Si estoy en Comida: desayuno+almuerzo+comida).
- **KCDT:** Kilocalorias que debería comer en esta comida concreta para este macronutriente concreto.
- **KMM:** Kilocalorias del macronutriente concreto del menú.
- **KMTC:** Kilocalorias totales que debería comer en esta comida.
- **ktm:** kilocalorias totales que tiene el menú a recomendar.

Reajuste del TMB

Durante el proyecto, se usa la fórmula para el cálculo del Metabolismo Basal, esto da como resultado las calorías que el usuario consume en un

día normal. A esta cantidad de calorías se le suele variar una cantidad de calorías que suele oscilar entre las 300 y 500 calorías. Para poder bajar o subir de peso de manera controlada (Aproximadamente 3 kilogramos al mes). Esta fórmula antiguamente estaba muy extendida entre nutricionistas y endocrinos a la hora de diagnosticar/proporcionar una dieta. Con los años, entro en desuso, debido a la generalidad que hacer la formula sobre las calorías que gasta cada persona, sin tener en cuenta metabolismos, posibles enfermedades, etcétera.

Desde el principio fue un problema que limitaba las posibilidades del proyecto. Por ello, se desarrolló un método que permite, a través de una colección de registros, llevar la cuenta de las variaciones de peso que sufre el usuario, y amoldar ese valor fijo, permitiendo su reajuste, y creando una fórmula que se adapta con mayor precisión a las necesidades del usuario.

Persistencia de los datos

No es propiamente un algoritmo, sino un recurso para solucionar una de las mayores limitaciones que presentaba usar una hoja de cálculo como base de datos y evitar datos sobre-escritos y diferentes problemas. Se procuró durante todo el programa atomizar al máximo posible cada método sobre la base de datos, haciendo que los cambios sean lo más 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.

Distinción entre tipos de comida

Uno de los retos a afrontar a lo largo de este proyecto fue 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 presentó diferentes técnicas a la hora de abordarlo.

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:

Se realizó un estudio sobre los diferentes métodos de distinción que se podían realizar. Se pensó en cadenas de caracteres (String) separadas por un delimitador cualquiera, y cargarlo en forma de cadena de caracteres separandolo utilizando algún delimitador, pero esto provocó ciertos problemas de carga y almacenamiento, problemas que aumentaban la complejidad del algoritmo al buscar a través de un bucle todas las posibilidades.

Se ideó 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 dígito creando una desigualdad en el programa. Por estas razones se decidió pasar al siguiente método, una mezcla de ambas, se iba a usar una cadena para calcular un único valor que cargar. Finalmente se pasó 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 cláusulas condicionales, para hacer la criba por tipo de comida.

5.4. 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, CSHARP...), entre los cuales se encontraba Python.

Por ejemplo, en asignaturas como Sistemas Inteligentes, Nuevas tecnologías, etc., o vistas también en la politécnica de Varsovia, en asignaturas como “Algoritmia y computación matemática”.

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

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 del proyecto, se usa una pizarra física, donde se colocan las diferentes tareas, con una fecha límite de realización, de esta manera, toda idea que se planteaba durante el desarrollo era plasmada en la pizarra y posteriormente hecha.

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 cómo 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. Si el tutor catalogaba una tarea, que se encontraba en "TO-VERIFY"; como incompleta pasaba de nuevo a "In progress" de esta forma, semanalmente se cumplían con una serie de propósitos de forma clara y concisa.

Este método fue estudiado en la asignatura de gestión de proyectos por el alumno, y culminado en su periodo de prácticas 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.

Investigación

De vital importancia es el nivel de capacidad de investigación y búsqueda de la información. 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 permite verificar de forma más precisa, tanto los datos o recursos necesarios para una investigación como la solución de errores que puedan emerger durante el desarrollo.

5.5. 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.

Psicología

Basándose en los principios de la psicología se busca la interacción adecuada con el usuario. Este proyecto no busca una forma rápida de llegar al cliente/usuario, sino constante. El cliente ha de convertir, las pautas de la aplicación, en su propio estilo de vida. Por ello se estudió las distintas formas de enseñanza, así como, el auge de la tecnología en el sistema educativo.

Aprender a aprender [6]

Usado en la educación primaria e infantil, es un método para enseñar al alumno, en base a que el alumno encuentre la propia respuesta. De esta manera desarrolla los estímulos positivos del alumno ante la adquisición de nuevos conocimientos.

La idea principal nace de las ventajas que tiene implementar un estilo de vida, sobre el seguir órdenes estrictas. Tras un periodo de investigación, se decantó por un estilo progresivo, 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, él mismo, se dé cuenta de que es lo mejor para él y aunque de manera lenta pero segura, llegue a la meta que tenga como objetivo.

Nutrición

Conocimientos básicos sobre el mundo de la nutrición, el reparto de los distintos nutrientes de cada alimento a lo largo del día y la correcta distribución de las calorías en base a las características del usuario.

Para el desarrollo del proyecto, se ha necesitado conocimientos sobre la calidad de los alimentos, así como, la predisposición de cada tipo de patología en base a las características de los alimentos. Logrando, a través del estudio e investigación, un análisis detallado para la veracidad de los datos a tratar.

Calidad de los alimentos

Parte fundamental del proyecto, la calidad marca el umbral por el cual se criban los alimentos. Este umbral es usado para el sistema de recomendaciones. Entre las posibles recomendaciones se encuentran las de umbral más bajo, complicando la mala elección por parte del usuario.

La calidad se mide con el **algoritmo Nutriscore**:

El algoritmo del semáforo o NUTRISCORE 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.

Interfaces Gráficas

Pese a tener los conocimientos sobre programación en Python, se desconocía cualquier tipo de implementación de interfaz gráfica. Debido a esto, gran parte 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.

5.6. Fases del proyecto

Las fases de este proyecto pueden ser divididas de diferentes modos, según qué 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 pulimento de la idea
- Desarrollo:
 - Creación de la estructura de los datos.
 - Carga y manejo de los datos
 - Creación del proyecto por interfaz de comando
 - Traducción del proyecto a interfaz gráfica
 - Pulimento de detalles y nuevas funcionalidades
- Pruebas
- Desarrollo de la documentación pertinente

Cronograma:

Tareas	Responsable	Fecha de inicio	Fecha final	Días	Estado
Pre-requisitos					
Perfeccionamiento de idea	Jesus Carro	11/20	11/30	10	Completado
Definir objetivos	Jesus Carro/Antonio Jesús Canepa	30-nov	03-dic	3	Completado
Inicio					
Determinar los requisitos	Jesus Carro	30-nov	03-dic	3	Completado
Requisitos de Hardware	Jesus Carro	03-dic	05-dic	2	Completado
Desarrollo					
Inicio					
Requisitos técnicos	Jesus Carro	03-dic	03-dic	0	Completado
Desarrollo de base de datos	Jesus Carro	05-dic	10-dic	5	Completado
Desarrollo Shell					
Interfaz de Usuario del Cliente	Jesus Carro	10-dic	08-mar	88	Completado
Prueba	Jesus Carro	08-mar	12-mar	4	Completado
Desarrollo Gráfico					
Completar desarrollo	Jesus Carro	12-mar	20-jun	100	Completado
Operaciones					
Prueba del sistema	Jesus Carro	20-jun	01-jul	11	Completado
Lanzamiento					
		05-jul	05-jul		

Figura 5.9: Tabla cronograma al completo

Trabajos relacionados

El mundo de la dieta como terapia para el tratamiento de distintas patologías se investiga desde hace años, tomando día a día más fuerza e importancia. Al principio se pensó que la dieta solo era efectiva para las patologías cuya causa estaba estrictamente vinculada a la alimentación del paciente, como es el caso de la diabetes. Con los años se fue fundamentando que se podrían prevenir otros tipos de enfermedades con una buena alimentación, como era el caso de los problemas cardiovasculares, la Osteoporosis e incluso el Cáncer. En la actualidad, numerosos estudios defienden que una adecuada alimentación no solo ayuda a prevenir y tratar enfermedades relacionadas con la alimentación, sino que es una parte fundamental del tratamiento y/o recuperación de enfermedades tanto livianas como importantes, como son la Obesidad, el Cáncer, y los problemas cardiovasculares.

6.1. Estudios:

Estudios DASH [2]

Los estudios DASH (Dietary Approaches to Stop Hypertension), son una serie de estudios relacionados para frenar la hipertensión a través una correcta alimentación, el primero de estos estudios se realizó en 1997. Realizado por la asociación del corazón americano (American Heart Association), el colegio americano de cardiología (American College of Cardiology), la sociedad contra el cáncer americano y muchas otros colegios y universidades.

A grandes rasgos se basa en la relación directamente proporcional entre comer una dieta rica en frutas y verduras y baja en azúcares, grasas y carnes

rojas y reducir la hipertensión y las enfermedades relacionadas con el corazón.

Las pruebas de grasas saturadas

Son una serie de estudios, que se basan en demostrar la correlación entre las grasas saturadas y el incremento de factores de riesgo en enfermedades relacionadas con el corazón. Pronto se observó en estudios como: <https://www.sevencountriesstudy.com/> del que hablaremos más adelante, que la calidad de vida de las personas cuya alimentación era alta en grasas saturadas era bastante peor, teniendo una tendencia superior al padecimiento de enfermedades cardiovasculares.

By 1965 at the latest, it was beyond a reasonable doubt that if you replace saturated fats with polyunsaturated fats, you get a substantial lowering of total cholesterol. por Katan

(Traducción: En 1965 se probó que si se remplazan las grasas saturadas por polinsaturadas se obtiene un decremento del colesterol total.)

El programa de prevención de diabetes (DPP)[16]

Actualmente uno de cada tres adultos tiene diabetes, lo más significativo es que 9 de cada 10 no lo saben.

Los resultados del DPP, se basaron en una prueba asignada de manera aleatoria a 3234 personas con diabetes que tomaron placebos, metformina (Una droga que baja el azúcar en sangre), o un “estilo de vida”.

La meta del estilo de vida consistía en perder peso y hacer ejercicio por lo menos dos horas y media a la semana. Para el momento en el que se realizaron fue un resultado increíble ver que el medicamento aumentaba en un 31 por ciento el riesgo de diabetes, mientras que el estilo de vida lo reducía en un 27 por ciento. Un estilo de vida había superado con creces los resultados del medicamento.

Pounds Lost

Estudio basado en la pérdida de peso, este estudio se realizó en 2004, y es uno de los estudios más largos sobre la dietoterapia realizados hasta la fecha. Consistía en observar entre 800 personas, que macronutriente habría que reducir para mejorar el estado físico y la salud, si grasas, proteínas o carbohidratos.

Fue un estudio que se realizó durante dos años, y se observó que los sujetos a observación habían adelgazado prácticamente lo mismo, y ninguno sufría ninguna deficiencia notable que indicase que un macronutriente era más necesario que otro.

Se esperaba que las personas con una dieta baja en carbohidratos y con resistencia a la insulina obtuvieran mejores resultados, pero no fue así. Entonces se llegó a la conclusión de que era porque todos estaban comiendo saludable, sin azúcares añadidos o hidratos refinados. Por lo que se demuestra que es increíble los avances que puede hacer la humanidad con una buena alimentación.

Seven countries Study [11]

Fue el primer gran estudio que se realizó para relacionar la calidad de vida con la dieta y los posibles factores de riesgo que podría tener una mala alimentación sobre la salud de las personas.

Nació en dos partes, primero se realizaron estudios en Italia, España, Sudáfrica y Japón de 1952 a 1956 y se sugirió, que los niveles de colesterol, las tasas de ataques cardíacos y las dietas variaban ampliamente y esto había que estandarizarlo de alguna manera, por ello en la “segunda parte”, se realizó un estudio piloto mas forma entre 1956 y 57 en Finlandia, Italia y Grecia que indicaban que sin duda había una correlación entre estos tres factores y que debía ser analizados de manera efectiva.

Esto fue solo el inicio del estudio, el posterior estudio, ya con las metas fijadas duró de 1958 a 1999, y también se dividió en dos partes:

1. Se realizaron encuestas sobre el estilo de vida y los factores de riesgo iniciales, y después de 5 y 10 años se realizó un seguimiento en 16 hombre de mediana edad de siete países.

2. La segunda fase se realizaron encuestas sobre la salud cardiovascular en ancianos de nueve cohortes europeas.

Este estudio proporciono evidencia de:

- El concepto de poblaciones sanas y enfermas
- Que los principales factores de riesgo cardiovasculares son Universales
- La hipótesis de la relación dieta-corazón
- Que la enfermedad cardiovascular es prevenible
- Que un estilo de vida saludable mejora varios ámbitos de la salud

6.2. Aplicaciones Similares

FatSecret

Aplicación Móvil y API de búsqueda, con una extensa base de datos, con toda la información nutricional de alimentos. Es muy usada en España para el seguimiento de dietas y el cálculo de macronutrientes diarios ayudando a miles de usuarios. Pero no te indica que has de comer, ni que es bueno ni nada, es útil exclusivamente para aquellas personas con un conocimiento básico sobre alimentación saludable.

DietPro

Es la aplicación más parecido al proyecto que se está detallando en este documento. Es una aplicación de pago, que crea un seguimiento de dietas con más de 3100 platos, y una gran variedad de alimentos, la cual, cambia en función a los objetivos y estados fisiológicos y patologías del paciente. Actualmente es una de las aplicaciones más potentes del mercado. El inconveniente es que se basa en dietas pautadas por expertos, teniendo el principal problema de toda aplicación existente relacionada con la nutrición, y es que, está basada en una dieta estricta, el usuario se puede hacer una idea de que es bueno o malo, pero sin saber muy bien porque, complicando el aprendizaje, creando necesidad de la aplicación, y el posible abandono al ser estricto.

MyFitnessPal

Aplicación bastante completa que además de ayudarte con tu alimentación con una base de datos, te ofrece una herramienta de gestión para tu actividad física, pero finalmente tiene el mismo problema que FatSecret, y es que necesitas de unos conocimientos básicos para que la aplicación sea realmente eficiente.

LifeSum

Planificador de comidas muy completo, que tiene una amplia gama de opciones entre las diferentes dietas y tipos, permitiendo llevar un gran seguimiento y ofreciéndote varias recetas entre las de la propia aplicación y su comunidad. El inconveniente es el mismo hablado hasta la fecha, y es que es un planificador, que no fomenta el aprendizaje y adquisición de un buen hábito de vida.

Conclusiones y Líneas de trabajo futuras

A continuación, se desarrollarán las diferentes conclusiones surgidas a lo largo de la creación del proyecto, a la par, de las líneas futuras de desarrollo que el proyecto a de seguir para la explotación a nivel tanto empresarial como informático.

7.1. Conclusiones

A continuación, se desarrollarán la conclusión que el proyecto ha ido dejando a lo largo de su creación:

- La dificultad de inculcar, enseñar o mostrar un camino a un estilo de vida más saludable a través de una aplicación para ordenador. En principio no se tienen ningún tipo de información, ni los recursos necesarios en cuanto a métodos pedagógicos, ni bases de datos que lo contemplen. Por lo que se partió de cero, basándonos en diseños, parcialmente similares, pero sin asemejarse demasiado a lo necesitado. Resulta un reto conseguir crear una aplicación que sirva para el aprendizaje o la enseñanza de un campo concreto.
- A lo largo de los años de estudio del alumno, se han realizado múltiples programas, o funciones, pero siempre eran cosas que: o ya existían y podías sacar información de diversas fuentes, o el profesor ponía al servicio del alumno, las herramientas necesarias para hacerlo. El verse envuelto en un escenario donde se debía crear de cero cosas que

no existían, fue un auténtico reto. La idea de estructurar primero una función o algoritmo, para ir desarrollándolo, cubriendo cualquier imperfección obviada durante el proceso, ayudo a valorar las facilidades que son otorgadas durante los años de estudio.

- El análisis de datos desde una hoja de cálculo. Se escogió una hoja de cálculo como base de datos, para facilitar el manejo y trato de los datos. A la larga acabó generando numerosas dificultades, generando diferentes problemas que no hubieran aparecido en casa de usar una base de datos real.
- Fue complicado la idea de desarrollar un proyecto desde cero, a lo largo de los estudios del alumno, se han realizado varios programas de distinta índole, pero siempre se partía de una base dada por el profesor y un guion a seguir. La idea de crear un proyecto, y tener que ir desarrollando y creando una serie de objetivos, fue complicado, pues muchas veces eran objetivos absurdos, imposibles, o que se desviaban totalmente del sino del proyecto.
- Crear un diseño desde cero, que fuera intuitivo, y de fácil navegabilidad fue un auténtico reto, teniendo que variar en ocasiones el diseño para mejorar la funcionalidad.
- Las limitaciones de las interfaces gráficas en Python. Como se ha nombrado en diversas ocasiones, las limitaciones que tienen dichas interfaces gráficas, y particularmente la librería Tkinter, ha resultado un gran inconveniente en varias etapas del proceso.
- El trabajo de investigación también fue complejo y extenso, sobre todo en el ámbito más médico-alimenticio donde existen diversas fuentes de información de las cuales muy pocas son fiables. Por ello se optó por una vía más personal como es la búsqueda de profesionales en la materia que pudiera apoyar los datos.

7.2. Líneas de trabajo futuras

A continuación, se plasmarán las posibles mejoras y adaptaciones del programa en caso de seguir con su desarrollo hasta llegar a una etapa final. Debido a la situación del proyecto enfocado, como un programa altamente funcional, del cual se espera un futuro desarrollo, no solo a nivel de aplicación, sino a nivel de proyecto empresarial; Se explicarán diferentes campos de ampliación del proyecto.

Separación modular y MVC

Pese a que la herramienta desarrollada, tiene su propio sistema organizativo, variante del modelo-vista-controlador, no deja de ser un método único, usado exclusivamente durante el desarrollo de este programa, por lo que existe una falta de estandarización, donde cualquier programador que quiera continuar el proyecto. Deberá estudiar el método de trabajo. Además, pese a que el programa se separa en varios módulos, debería ser posible separarlo en más módulos diferentes, más específicos y no exclusivamente en los tres módulos principales usados en el proyecto (Sin contar el Main).

Base de datos en Servidor

La idea de una aplicación con una amplia base de datos de alimentos, y una serie de datos sobre los usuarios y sus respectivas patologías, no puede ser tratada de manera local. Debido a la ley de protección de datos [1], los datos de carácter médico, son datos muy restrictivos y de alta importancia, penados con cárcel.

Por ello se implantaría una base de datos en un servidor On-line altamente protegido para evitar cualquier tipo de problemas. Además de lograr la expansibilidad de la base de datos.

Se parte de una pequeña muestra, de todos los posibles menús y patología que se podrían añadir a la base de datos. No deja de ser un Start-Up de un gran desarrollo, con los ejemplos básicos para la muestra de su funcionamiento.

Perfeccionamiento del algoritmo de recomendación

Pese a ser el eje central de la aplicación es un algoritmo todavía a perfeccionar. Ajusta en base a las necesidades del usuario, a la situación del usuario en ese momento, y a las características del alimento, una variable que servirá de "peso", para mostrar las recomendaciones más adecuadas. No obstante, este algoritmo da mucho peso a las características del alimento y las necesidades del usuario en ese momento, sobre las necesidades del usuario de todo el día. Y si dos alimentos de distinta calidad tienen una puntuación similar, se recomendará el de mejor puntuación sin tener en cuenta la calidad.

Adaptación a diferentes plataformas

Hoy en día las plataformas multimedia y los distintos dispositivos "smart", son cada vez más utilizados en el ámbito de la enseñanza. Siendo una

metodología de educación cada vez más aceptada. Dentro de las nuevas tecnologías, las Tablets y los SmartPhones, son las herramientas con mayor crecimiento en cuanto a su uso se refiere; Por ello, la idea de una aplicación interactiva, que enseñe al usuario a habitar un estilo de vida saludable, ha de ser una aplicación que esté al alcance del usuario de la forma más sencilla posible. La ampliación de este proyecto a distintas plataformas que mejore la comunicación con el usuario es en parte una obligación.

Bibliografía

- [1] Grupo Adaptalia. Ley de protección de datos médicos. 2019.
- [2] American Cancer Society American Heart Association, American College of Cardiology. The dash studies. 1997.
- [3] autoaprendizaje7. Qué es y como calcular la tasa metabólica basal, 2010. [Internet; descargado 16-junio-2019] <https://autoaprendizaje7.wordpress.com/2010/12/24/el-aprendizaje-autonomo-es-mas-efectivo-cuando-reconoces-que-es-lo-importante->
- [4] Beatriz. El uso de las apps en educación. 2014.
- [5] Consuelo Boticario. ¿una alimentación sana puede prevenir el cáncer?e, 2005. [Internet; descargado 15 de Junio de 2019] <http://www.analesranf.com/index.php/aranf/article/view/206>.
- [6] Enrique Ballester Sarrias. Angel Montes Carlos Camiña Catala, Angel Cuenca Lacruz. Autoaprendizaje y formación en cualidades mediantelas herramientas multimedia. 2015.
- [7] competenciasbasicascordoba. Competencia y actitudes para seguir aprendiendo de forma autónoma a lo largo de la vida, 2018. [Internet; consultado 15 de Junio de 2019] <https://competenciasbasicascordoba.webnode.es/aprender-a-aprender/>.
- [8] Josefina del prado. Prevención primaria, secundaria y terciaria, 2013. [Internet; descargado 10-junio-2019] <https://blogs.imf-formacion.com/blog/prevencion-riesgos-laborales/actualidad-laboral/prevencion-primaria-secundaria-y-terciaria/>.

- [9] José Luis Ulizarna García. Tecnologías multimedia en el ámbito educativo, 2017. [Internet; descargado 10-junio-2019] <https://recyt.fecyt.es/index.php/pixel/article/viewFile/61110/37124>.
- [10] health Keeper. Qué es y como calcular la tasa metabólica basal, 2016. [Internet; descargado 23-junio-2019] <https://www.axahealthkeeper.com/blog/que-es-y-como-calcular-la-tasa-metabolica-basal/>.
- [11] Daan Kromhout. Qué es y como calcular la tasa metabólica basal, 2012. [Internet; descargado 25-junio-2019] <https://www.sevencountriesstudy.com/>.
- [12] The Lancet. Measuring progress from 1990 to 2017 and projecting attainment to 2030 of the health-related sustainable development goals for 195 countries and territories: a systematic analysis for the global burden of disease study 2017, 2018. [Internet; consultado 25-junio-2019] [https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(18\)32281-5/fulltext](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(18)32281-5/fulltext).
- [13] Wes McKinney. *Python for data Analysis*. O'REILLY, 2017.
- [14] Periodico El Mundo. El 81 [Internet; consultado 25-junio-2019] <https://www.elmundo.es/salud/2014/02/12/52fb7d5022601d7f228b4570.html>.
- [15] Dr. Pedro Pinheiro. Obesidad calorías para adelgazar 1 kg por semana, 2019. [Internet; descargado 25-junio-2019] <https://www.mdsaude.com/es/obesidad/calorias-para-adelgazar>.
- [16] The Diabetes Prevention Program. Diabetes prevention program. 2002.