Optimeal

# Resumen

Optimeal es una aplicación desarrollada en Android Studio usando Kotlin como lenguaje. La aplicación crea una experiencia personalizada en la planificación de menús semanales. Nuestro principal objetivo es que sea modificable al máximo, así lo deja claro nuestro slogan “Optimeal, tu chef personalizado”. Los usuarios introducen datos como macros, alergias, preferencias, límites de gasto, etc. y la aplicación genera semanalmente menús que cumplen con sus requisitos. Una vez generados los menús el propio usuario puede ir variando los platos asignados si así lo desea.

La aplicación tiene otros apartados como recetas, que vienen de una API; lista de la compra, que se autogenera en el momento de la creación del menú.

# Justificación

Debido a la falta de tiempo de las nuevas generaciones para la planificación y preparación de menús semanales se está notando que la alimentación de estos está empeorando en contraposición a sus antecesores.

Diversos estudios, como el de la EIT Food, afirman que la falta de educación en los centros, las pocas políticas destinadas a ello y la industria que cada día aboga mas por el ultra-procesado, han hecho que los jóvenes, preocupados por su alimentación, busquen en redes sociales y aplicaciones la solución a un problema que nadie se presta a dar (Tech, 2021).

De ahí el nacimiento de esta idea, destinada a ayudar a las nuevas generaciones a cuidar su alimentación sin necesidad de mucho conocimiento nutricional ni gastos de tiempo y dinero en ello. Optimeal tiene como propósito principal hacer la vida más fácil a una generación de jóvenes cada vez más preocupados por su alimentación y con cada vez menos tiempo para pensar en ella.

# Análisis de mercado

Vamos a desarrollar esta parte mediante la creación de un DAFO, donde veremos Debilidades, Amenazas, Fortalezas y Oportunidades.

## Debilidades

## Amenazas

* Servicios de nutrición profesional
* La mayoría de los recursos como APIs contienen platos de cocinas extranjeras

## Fortalezas

* Conocimiento de desarrollo de ambos socios fundadores.
* Formación en nutrición y cocina de uno de los socios durante ocho años.
* Conocimiento interno de diseño de interfaces.

## Oportunidades

* Falta de tiempo de las nuevas generaciones para la planificación y la cocina.
* Falta de conocimiento culinario de las nuevas generaciones.
* Gran preocupación tanto de las entidades dedicadas a la alimentación como de los propios ciudadanos.

# Requisitos

* Login
  + Algoritmo BMR y macros
  + Usuario Administrador
* Creación de menús personalizados
  + Algoritmo
* Base de datos usuarios con macros
* Base de datos recetas con macros
* Mi lista de la compra

# Metodología

La organización desempeña un papel fundamental en el desarrollo de proyectos, siendo determinante para el éxito de éste. La efectividad de un proyecto reside en la estructuración del equipo y la gestión ordenada de las diversas fases involucradas. Para lograr una planificación y ejecución eficientes, es crucial seleccionar la metodología más apropiada con el fin de ordenar, planificar y optimizar la gestión del proyecto. A continuación, explicamos 3 de las principales metodologías más usadas.

1. Metodología Ágil:

Considerada más como un principio de gestión que una metodología técnica, destaca por su enfoque iterativo y flexible.

Este enfoque permite gestionar eventos y cambios imprevistos de manera más efectiva. Adecuada para casi todos los equipos, la clave está en decidir qué metodología combinar con ella para avanzar según las necesidades del cliente o del equipo.

1. Metodología KanBan:

La metodología Kanban se centra en representar visualmente las tareas pendientes de un proyecto mediante tableros. Los equipos ágiles adoptan este enfoque para mejorar la visualización de flujos de trabajo y el progreso del proyecto, evitando posibles cuellos de botella.

Recomendado para equipos de todos los tamaños, los tableros Kanban son especialmente útiles para equipos remotos al facilitar la visualización y el seguimiento del trabajo, independientemente de la ubicación de los miembros del equipo.

1. Metodología Scrum:

La metodología Scrum se caracteriza por utilizar "sprints" breves para estructurar los ciclos de proyectos, con una duración de una a dos semanas.

Scrum destaca por su enfoque único, una gestión que se guía de reuniones diarias, demostraciones, “sprints” y análisis retrospectivos, con el objetivo de mantener la conexión entre los participantes del proyecto y asegurar la finalización puntual de las tareas.

Aunque Scrum es una metodología de gestión de proyectos en sí misma, se asocia estrechamente con el enfoque ágil debido a principios compartidos, como la valoración de las personas y la colaboración de los equipos sobre los procesos. Se recomienda especialmente para equipos que emplean un enfoque ágil.

Para llevar a cabo reuniones diarias efectivas, planificar sprints y realizar análisis retrospectivos significativos, es esencial contar con un medio que permita visualizar el trabajo en sus diversas etapas y realizar un seguimiento detallado de todas las tareas en curso.

Por ello, la metodología que usaremos será la llamada ScrumBan, que utiliza los tableros Kanban para gestionar las tareas pendientes de los sprints y organizar el flujo de trabajo durante cada ciclo de Scrum, asegurando el éxito del proceso.

Diagrama

Descripción generada automáticamente Diagrama

Descripción generada automáticamente con confianza baja

Programación

# Algoritmo Menús

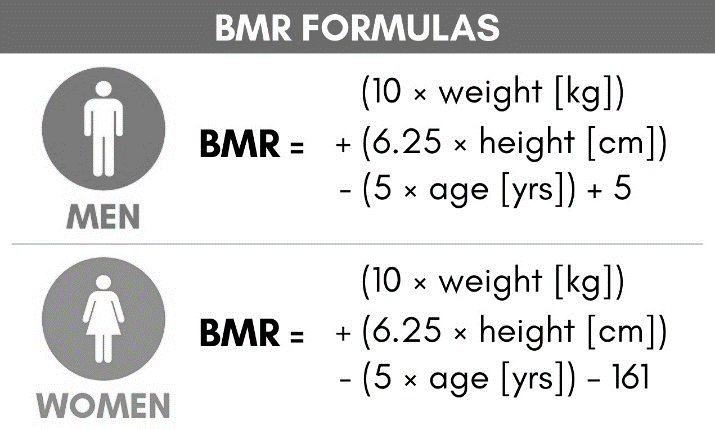
### Extracción del TDEE(Total daily Energy Expenditure)

#### BMR

<https://www.escuelaculturismonatural.com/calculo-del-tdee/>

El proceso para el calculo de la tasa de gasto energético diario de nuestros usuarios incluye dos pasos: el calculo del BMR(Tasa metabolismo basal) y posteriormente el cálculo del TDEE.

El BMR, a diferencia del TDEE, son las kcal que nuestro cuerpo consume diariamente sin tener en cuenta nuestra actividad física ni nuestros objetivos. Se calcula mediante la formula de *Harris Benedict:*



Como podemos observar solo tiene en cuenta datos puramente anatómicos, es decir lo calcula mediante el peso, la altura, el sexo y la edad.

Una vez que tenemos el BMR del usuario, por ejemplo el de un hombre de 31 años que mide 180cm y pesa 70kg:

*(10\* 70) + (6.25\*180) – (5 \* 31) +5 = 1675kcal/diarias*

#### Activity Factor

Podemos calcular el TDEE teniendo en cuenta los parámetros de actividad física y objetivos del usuario. Para lo siguiente contamos con dos tablas, dependiendo de la actividad y objetivos sacamos dos multiplicadores que luego usaremos junto con el BMR para calcular la ingesta personalizada del usuario.



Aquí observamos que dependiendo del nivel de ejercicio que el usuario haga el multiplicador varía progresivamente. Siguiendo con el ejemplo del hombre de 31 años, vamos a suponer que hace un ejercicio moderado, de 3 a 5 dias a la semana, por lo que multiplicaremos el BMR \* 1.55.

*BMR \* ActivityFactor->*

*1675\* 1.55= 2596’25 kcal/diarias*

Vemos como ha aumentado la ingesta de calorías, es lógico, puesto que el BMR calcula las calorías en base a necesidades básicas. Una vez que se le pide un gasto de energía extra al cuerpo por poco que sea las calorías han de aumentar.

#### Objetivos

Al igual que con el nivel de actividad los objetivos del usuario, ya sea ganar peso o perder masa muscular, también cuentan en la ingesta calórica diaria. Utilizamos también un multiplicador al que llamamos déficit.

Si el usuario quiere perder peso multiplicamos las calorías diarias por 0.8, reduciendo así la ingesta de las mismas. Si por el contrario lo que quiere es aumentar de peso multiplicamos la ingesta por 1.07.

El usuario también tiene la opción de no querer ni perder peso ni ganarlo, por lo que no utilizamos este multiplicador.

Continuemos con el ejemplo anterior, suponiendo que el objetivo del hombre de 31 años es perder peso, finalizando así el cálculo del TDEE:

*BMR\* ActivityFactor\* Déficit->*

*2596.25\* 0.8= 2.077kcal/diarias*

Después del calculo del BMR y su producto por actividad física y objetivos tenemos la ingesta calórica diaria a la que se debe someter nuestro usuario. Este dato lo guardamos en nuestra base de datos para usarlo todas las semanas al generar el nuevo menú.

### Extracción de los macronutrientes

https://www.womenshealthmag.com/es/nutricion-dietetica/a36590862/calcular-macros/

Los macronutrientes que vamos a controlar son las grasas, las proteínas y los carbohidratos. Por lo general las grasas y las proteínas se mantienen iguales para todos los usuarios, siendo la grasa en torno al 30% y la proteína el 20%.

Sin embargo, los carbohidratos han de cambiar dependiendo de los objetivos del cliente. Siendo más altos en usuarios que quieren ganar peso y más bajo en usuarios que quieren perderlo. La ingesta que se recomienda en dietas que quieren perder peso es del 40% de carbohidratos mientras que en dietas donde el objetivo es ganar peso pueden aumentar hasta el 50%.

El siguiente paso es extraer los gramos de estos macronutrientes desde las kcal. Para ello tenemos tres fórmulas que seguir:

1. **Grasas (de kcal a gr)**

Cada gramo de grasa aporta al cuerpo 9 calorias, la formula es:

En nuestro ejemplo serían :

*2077kcal \* 0.3 / 9= 69.23 gr grasa/día*

1. **Proteínas (de kcal a gr)**

Debemos tener en cuenta que cada gramo de proteína constituye 4 calorías, por lo que la fórmula es la siguiente:

En nuestro ejemplo serían :

*2077kcal \* 0.2 / 4= 103.85 gr proteína/día*

1. **Carbohidratos (de kcal a gr)**

En el caso de los carbohidratos cada gramo nos da 4 calorías, por lo que:

En nuestro ejemplo serían:

*2077kcal \* 0.4 / 4= 207.7 gr carbohidratos/día*

En resumen nos quedarían divididas las kcal en gramos de macronutrientes de la siguiente manera:

* Kilocalorías: 2077 kcal
* Grasas: 69.23 gr al día
* Proteínas: 103.85 gr al día
* Carbohidratos: 207.7 gr al día

<https://estilosdevidasaludable.sanidad.gob.es/alimentacionSaludable/queSabemos/enLaPractica/distribuir/diario/home.htm>

Pantalla de un video juego

Descripción generada automáticamente con confianza media

|  |  |  |  |
| --- | --- | --- | --- |
|  | Grasa | Proteína | Carbohidratos |
| Desayuno | 33% | 33% | 30% |
| Comida | 33% | 33% | 60% |
| Cena | 33% | 33% | 10% |

Siguiendo el ejemplo nos quedaría una tabla como la siguiente:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | kcal | Grasa | Proteína | Carbohidratos |
| Desayuno | 519 | 22.85 g | 34.27 g | 62.31 g |
| Comida | 830.8 | 22.85 g | 34.27 g | 124.62 g |
| Cena | 726.95 | 22.85 g | 34.27 g | 20.77 g |
| Total | 1976 kcal | 69.23 gr | 103.85 gr | 207.7 gr |

<https://www.ucm.es/data/cont/docs/458-2019-01-04-Guia-Practicas-2019-web.pdf>

|  |  |  |  |
| --- | --- | --- | --- |
|  | kcal | Grasa | Proteína |
| Desayuno | 519 | 129.75-181.65 | 77.85-155.7 |
| Comida | 830.8 | 207.7-290.78 | 124.62-249.24 |
| Cena | 726.95 | 181.74-254.43 | 109.04-218.9 |

### División de las calorías diarias

Una vez que tenemos las calorías diarias que ha de consumir nuestro usuario las vamos a repartir siguiendo la siguiente lógica

1. **Desayuno (25% de las calorías diarias):**
   * Por la mañana, el cuerpo necesita energía para comenzar el día y activar el metabolismo después del ayuno nocturno.
   * Al asignar el 25% de las calorías diarias al desayuno, se proporciona una cantidad adecuada de energía para impulsar las actividades matutinas y mantener la concentración y el rendimiento mental.
   * Un desayuno equilibrado debe incluir carbohidratos saludables, proteínas magras y grasas saludables para una energía sostenida y una sensación de saciedad.
2. **Comida (almuerzo) (40% de las calorías diarias):**
   * El almuerzo se sitúa en el punto medio del día y es una oportunidad para reponer energías y nutrientes después de la actividad matutina.
   * Al asignar el 40% de las calorías diarias al almuerzo, se proporciona una cantidad sustancial de energía para mantener el rendimiento físico y mental durante la tarde.
   * El almuerzo también es importante para evitar la fatiga y el hambre excesiva más tarde en el día, lo que puede llevar a comer en exceso en la cena.
3. **Cena (35% restante de las calorías diarias):**
   * La cena es la última comida del día y generalmente precede a un período de descanso y sueño.
   * Al asignar el 35% restante de las calorías diarias a la cena, se garantiza que se proporcione suficiente energía para mantener las funciones corporales durante el descanso nocturno, pero sin sobrecargar el sistema digestivo antes de dormir.
   * Es importante optar por opciones de alimentos más livianas y fáciles de digerir en la cena para promover un sueño reparador y evitar problemas digestivos.

Haciendo esta división para nuestro ejemplo de 2077kcal diarias quedarían repartidas así:

|  |  |
| --- | --- |
| Desayuno | 519 kcal |
| Comida | 830,8 kcal |
| Cena | 726,95 kcal |

Desglosemos ahora los macronutrientes

|  |  |  |  |
| --- | --- | --- | --- |
|  | Grasa | Proteína | Carbohidratos |
| Desayuno | 33% | 33% | 30% |
| Comida | 33% | 33% | 60% |
| Cena | 33% | 33% | 10% |

<https://www.yummly.com/recipe/Strawberry-Pretzel-Salad-9024267>

<https://www.prospre.io/features>

<https://www.foodspring.es/magazine/recetas-fitness/risotto-de-calabaza-y-pollo>

<https://csvjson.com/csv2json>

# Tecnologías usadas

### RecyclerView

RecyclerView simplifica la presentación eficiente de conjuntos extensos de datos al proporcionar una estructura donde puedes suministrar los datos y definir la apariencia de cada elemento. La biblioteca RecyclerView, a su vez, genera dinámicamente los elementos según la demanda.

Conforme a su designación, RecyclerView recicla los elementos individuales. Cuando un elemento sale de la pantalla, en lugar de destruir su vista, RecyclerView la reutiliza para los nuevos elementos que están siendo desplazados y ahora se encuentran en la pantalla. Esta metodología contribuye significativamente a mejorar el rendimiento y la capacidad de respuesta de la aplicación, al mismo tiempo que reduce el consumo energético.

**Clases Clave**

Diversas clases desempeñan roles interconectados para construir tu lista dinámica.

* **RecyclerView** actúa como el ***ViewGroup*** que alberga las vistas correspondientes a tus datos. Siendo en sí misma una vista, incorporas RecyclerView a tu diseño de la misma manera que añadirías cualquier otro componente de la interfaz de usuario.
* Cada elemento individual de la lista se define mediante un objeto contenedor de vistas. Inicialmente, este contenedor carece de datos asociados. Luego de su creación, RecyclerView lo enlaza con los datos pertinentes. Para definir dicho contenedor de vistas, es necesario extender ***RecyclerView.ViewHolder***.
* La RecyclerView solicita estas vistas y las vincula a los datos a través de llamadas a los métodos del adaptador. La definición del adaptador requiere que extiendas ***RecyclerView.Adapter***.
* El administrador de diseño organiza los elementos individuales de tu lista. Todos los administradores de diseño se fundamentan en la clase abstracta ***LayoutManager*** de la biblioteca.

Utilizaremos el recycler view dentro de un fragment, usando así su viewModel, la estructura de las carpetas será la siguiente:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteVemos que tenemos una carpeta adapter, que contiene la clase HomeAdapter y la clase HomeViewHolder explicadas anteriormente.

Tendremos la clase HomeModel, que será el modelo que van a seguir los elementos del recycler view.

Y por último tendremos el HomeFragment con su ViewModel correspondiente.

A continuación, explicaré cada clase y las funcionalidades que cumple:

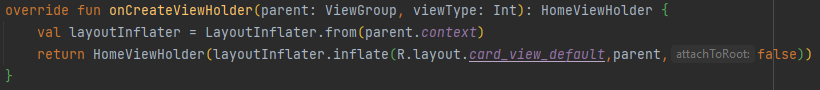
***HomeAdapter***



Esta clase nos permite recoger un simple listado y convertirlo en un recyclerView, recogiendo por parámetro una lista de IngredientModel

Esta clase va a extender de RecyclerView.Adapter, indicándole la clase de nuestro viewHolder, que en este caso es HomeViewHolder.

Se implementan automáticamente los métodos abstractos de RecyclerView



Este método devuelve el viewHolder, diciendo el layout que va a modificar.

Va a devolver ese ítem que acabamos de crear al viewHolder por cada objeto que haya en la lista de IngredientModel.

Texto

Descripción generada automáticamente

Recorre cada uno de los ítems llamando al método render creado en el HomeViewHolder



Y por último este método devuelve la cantidad de ítems que hay en la lista de IngredientModel.

***HomeViewHolder***



Esta clase extiende de ViewHolder, para decirle que va a ser de tipo ViewHolder



Aquí creo el binding, para poder acceder a los ids de la vita deseada, en este caso de la vista de card\_view\_default.xml

Texto

Descripción generada automáticamente

Y por último creo el método render, que recibirá el IngredientModel desde la calse HomeAdapter, y actualizo la vista xml con los sus datos.

correspondiente

# Requisitos

# RFTP

# Bibliografía

Tech, R. T. F. (2021, 13 septiembre). *Generación Z, la más preocupada por mejorar su alimentación*. THE FOOD TECH - Medio de noticias líder en la Industria de Alimentos y Bebidas. https://thefoodtech.com/tendencias-de-consumo/generacion-z-la-mas-preocupada-por-mejorar-su-alimentacion/

<https://developer.android.com/guide/topics/ui/layout/recyclerview?hl=es-419>

Necesito que me crees un menu del dia con desayuno comida y cena que tenga las sigugientes caracterisitcas:

Desayuno:

-alrededor de 519kcal

-entre 129 y 181 gr de grasa

-entre 77 y 155 gr de proteina

-alrededdor de 61 gr de carbohidratos

Comida:

-alrededor de 830kcal

-entre 207 y 290 gr de grasa

-entre 124 y 250 gr de proteina

-alrededdor de 124 gr de carbohidratos