

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

**Diseño de una aplicación para la optimización de dietas
mediante el uso de inteligencia artificial.**

Fabrizzio Andree Uscocovich Galarza

Ingeniería en Ciencias de la Computación

Trabajo de fin de carrera presentado como requisito
para la obtención del título de
Ingeniero en Ciencias de la Computación

Quito, 12 de mayo de 2025

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ**Colegio de Ciencias e Ingenierías****HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA****Diseño de una aplicación para la optimización de dietas personalizadas
mediante el uso de inteligencia artificial.****Fabrizio Andree Uscocovich Galarza****Nombre del profesor, Título académico****Jorge Ricardo Flores Moyano, Ph.D**

Quito, 12 de mayo de 2025

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos: Fabrizzio Andree Uscocovich Galarza

Código: 323115

Cédula de identidad: 1311732646

Lugar y fecha: Quito, 12 de mayo de 2025

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en <http://bit.ly/COPETheses>.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on <http://bit.ly/COPETheses>.

RESUMEN

El estilo de vida moderno impulsado por los avances tecnológicos ha transformado el modo de vivir de las personas en la actualidad tanto para bien como para mal. A pesar de simplificar un gran número de tareas, también ha fomentado el incremento de hábitos poco saludables como puede ser el sedentarismo. En particular, el incremento de cosas como el trabajo remoto, ayudado por el fácil acceso a comidas rápidas y procesadas, ha contribuido al aumento de enfermedades como la obesidad, diabetes e hipertensión. Para intentar contrarrestar estos efectos negativos, este proyecto propone el desarrollo de una aplicación basada en inteligencia artificial que brinde recomendaciones dietéticas personalizadas, considerando características individuales de los usuarios. Con este enfoque se busca convertir la tecnología en una aliada para la promoción de hábitos saludables. Entre los resultados esperados está el mejoramiento de la calidad de vida de los usuarios, brindándoles una alimentación más equilibrada y adaptada a sus necesidades. En un futuro, esta aplicación podría ser mejorada incorporando cosas como seguimiento en tiempo real o mejor integración con dispositivos de salud como glucómetros.

Palabras clave: Inteligencia artificial, sedentarismo, hábitos saludables, calidad de vida, enfermedades

ABSTRACT

The modern lifestyle, driven by technological advances, has transformed the way people live today, both for better and worse. Despite simplifying a large number of tasks, it has also fostered the rise of unhealthy habits such as a sedentary lifestyle. In particular, the increase in remote work, aided by easy access to fast and processed foods, has contributed to the rise of diseases such as obesity, diabetes, and hypertension. To counteract these negative effects, this project proposes the development of an artificial intelligence-based application that provides personalized dietary recommendations, considering the individual characteristics of users. This approach seeks to turn technology into an ally in promoting healthy habits. Among the expected results is an improvement in users' quality of life by providing them with a more balanced diet tailored to their needs. In the future, this application could be improved by incorporating features such as real-time tracking or better integration with health devices like glucometers.

.

Keywords: Artificial intelligence, sedentary lifestyle, healthy habits, quality of life, diseases

TABLA DE CONTENIDO

Introducción	10
Desarrollo del Tema.....	12
Conclusiones.....	41
Referencias bibliográficas.....	43
Anexo A: Título	46
Anexo B: Título	47
Anexo C: Título	48

ÍNDICE DE TABLAS

TABLA 1: RESUMEN DE LA DISPOSICIÓN DEL DATASET	17
TABLA 2: MEJORES PARAMETROS DECISION TREE	22
TABLA 3: REPORTE DE CLASIFICACIÓN DECISION TREE	22
TABLA 4: MEJORES PARÁMETROS - RANDOM FOREST	24
TABLA 5: REPORTE DE CLASIFICACIÓN - RANDOM FOREST	24
TABLA 6: REPORTE DE CLASIFICACIÓN - MLP	28
TABLA 7: SELECCION DE LAS 10 MEJORES CARACTERISTICAS	31
TABLA 8: REPORTE DE CLASIFICACIÓN DEL MODELO MLP CON FS	32
TABLA 9: TABLA DE METODOS UTILIZADOS POR LA APLICACIÓN	38

ÍNDICE DE FIGURAS

FIGURA 1: PIPELINE DEL PROYECTO	14
FIGURA 2: DISTRIBUCIÓN VARIABLE OBETIVO	19
FIGURA 3: RANKING DE CARACTERISTICAS RF.....	20
FIGURA 4: MATRIZ DE CONFUSIÓN DT.....	23
FIGURA 5: MATRIZ DE CONFUSIÓN RF	25
FIGURA 6: CURVAS DE VALIDACIÓN RF	26
FIGURA 7: GRAFICAS LOSS NEURAL NETWORK.....	28
FIGURA 8: CARACTERISTICAS MUTUAL_INFO	30
FIGURA 9: CARACTERISTICAS RANDOM FOREST.....	31
FIGURA 10: MATRIZ DE CONFUSIÓN MLP-FS	33
FIGURA 11: VISUALIZACION ESPACIO CON PCA.....	34
FIGURA 12: ARQUITECTURA DE LA APLICACIÓN	36
FIGURA 13: DIAGRAMA DE FLUJO DE LA APP.....	37
FIGURA 14: JSON ENVIADO AL API FLASK.....	38
FIGURA 15: IMAGEN DE LA APP EN SIMULADOR	39
FIGURA 16: RECOMENDACIÓN AL USUARIO	40

INTRODUCCIÓN

El desarrollo de nuevas tecnologías como la inteligencia artificial ha abierto oportunidades significativas en campos como la salud y la nutrición, donde persisten desafíos relacionados con la adopción de hábitos alimenticios saludables. En este contexto, surge la necesidad de soluciones que integren personalización, precisión y facilidad de uso, especialmente en dispositivos móviles orientados al bienestar.

Este proyecto propone el diseño de una aplicación inteligente capaz de recomendar planes alimenticios personalizados, adaptados a las características clínicas y nutricionales de cada usuario. A partir de un enfoque basado en aprendizaje automático, la aplicación analiza información relevante sobre el estado de salud y los hábitos alimenticios para sugerir dietas alineadas con los requerimientos individuales.

El desarrollo se apoya en un conjunto de datos estructurado que incluye variables clave como el tipo de condición médica crónica (por ejemplo, diabetes, hipertensión u obesidad), el consumo de sodio y azúcar, el nivel de actividad física del usuario, y el nivel de severidad de su condición médica. Estas características fueron seleccionadas cuidadosamente para garantizar un equilibrio entre efectividad predictiva y simplicidad en la interacción con el sistema.

La aplicación está diseñada para ser intuitiva y accesible, reduciendo al mínimo la carga de información requerida al usuario sin comprometer la calidad de las recomendaciones. Este enfoque busca no solo fomentar una mejor adherencia a pautas alimenticias saludables, sino también facilitar su integración en contextos reales a través de una experiencia de usuario eficiente y personalizada.

Objetivo General:

Diseñar e implementar una aplicación móvil multiplataforma basada en inteligencia artificial que recomiende planes alimenticios, considerando condiciones clínicas del usuario como diabetes, hipertensión, obesidad y otras, con el fin de mejorar su calidad de vida y fomentar hábitos saludables.

Objetivos Específicos:

- Construir un dataset clínico personalizado a partir de literatura científica confiable, que relacione condiciones médicas crónicas con recomendaciones dietéticas validadas.
- Aplicar técnicas de machine learning, incluyendo Decision Tree, Random Forest y redes neuronales MLP, para predecir la dieta óptima según las características del usuario.
- Implementar un proceso de selección de características para reducir la complejidad del modelo sin perder precisión, usando información mutua y análisis de importancia.
- Desarrollar una interfaz móvil con Flutter que recoja los datos clínicos del usuario de manera intuitiva y los envíe al backend de predicción.
- Integrar una API en Flask que reciba datos desde la app, realice la predicción usando el modelo entrenado y devuelva un plan alimenticio personalizado.
- Mostrar al usuario un menú completo diario (desayuno, almuerzo y cena) basado en la dieta recomendada, utilizando archivos estructurados en formato JSON.

DESARROLLO DEL TEMA

El uso de tecnologías digitales e inteligencia artificial IA en el ámbito de la nutrición ha generado oportunidades significativas para mejorar la adherencia a dietas terapéuticas en poblaciones con enfermedades crónicas. Si bien existe una amplia oferta de aplicaciones móviles orientadas a la promoción de hábitos saludables, estas suelen estar dirigidas a personas sanas, enfocándose en metas generales como la pérdida de peso o el bienestar físico, sin considerar variables clínicas específicas como la presencia de diabetes, hipertensión u obesidad.

La mayoría de aplicaciones de recomendación dietética actualmente disponibles en tiendas móviles (Google Play, App Store) priorizan enfoques generalistas, centrados en calorías, macronutrientes o preferencias alimenticias. Aunque algunas incluyen funcionalidades para seguimiento de indicadores como glucosa o presión arterial, pocas integran dichas variables como elementos centrales del sistema de recomendación. Esta brecha limita su aplicabilidad clínica, especialmente en pacientes con requerimientos dietéticos específicos definidos por guías médicas.

Estudios recientes han comenzado a abordar esta carencia mediante el desarrollo de aplicaciones diseñadas explícitamente para usuarios con condiciones crónicas. Una revisión conducida por Elicit sobre aplicaciones para obesidad, diabetes e hipertensión recopiló cuatro investigaciones destacadas:

- Callejas Gil et al. desarrollaron un sistema de monitoreo remoto mediante app móvil y web, aplicado a pacientes con obesidad, diabetes, hipertensión y dislipidemia. Aunque el grupo de intervención tuvo menor adherencia (14%) que el control (79%), se observaron mejoras modestas en peso, presión arterial y hemoglobina glicosilada [1].
- Franco et al. emplearon una plataforma web interactiva para mejorar el control de factores de riesgo cardiovascular en pacientes con hipertensión y diabetes mal controladas. El estudio evidenció mejoras en la adherencia terapéutica, presión arterial y perfil lipídico [2].
- McKensy-Sambola et al. propusieron un sistema de recomendación basado en ontologías nutricionales, dirigido al manejo de obesidad. El sistema logró una precisión

del 87% en la recomendación de dietas, demostrando su alineación con expertos humanos [3].

- Villar Jiménez et al. implementaron sesiones de teleorientación para pacientes con hipertensión y diabetes, mejorando la adecuación de la ingesta alimentaria del 6.25% al 71.25%, lo cual refuerza el potencial de intervenciones remotas y personalizadas [4].

Estos trabajos evidencian una tendencia creciente hacia la integración de herramientas digitales en el cuidado nutricional, aunque aún son limitadas las aplicaciones que integran explícitamente variables clínicas.

El proyecto realizado se distingue de las soluciones anteriores al incorporar como variables predictoras más condiciones médicas crónicas del paciente, integrando directamente factores como diabetes, hipertensión, obesidad, enfermedad renal y enfermedades del corazón en la arquitectura del modelo de recomendación. Esta orientación nos permite generar planes alimenticios adaptados con mayor precisión a las necesidades medicas del usuario.

Además, el proyecto se apoya en un pipeline técnico sólido que incluye:

- Modelos avanzados de clasificación como Random Forest y Redes Neuronales MLP.
- Selección de características mediante información mutua para optimizar la eficiencia del sistema.
- Interfaz multiplataforma en Flutter integrada con un backend en Flask para escalabilidad y facilidad de implementación.

Este enfoque proporciona una solución técnica robusta y clínicamente relevante, que responde a una necesidad no completamente cubierta por aplicaciones actuales en el mercado. La integración de IA con evidencia médica y diseño centrado en enfermedades representa una contribución innovadora al estado del arte en salud digital y nutrición personalizada. A continuación se presenta el pipeline de los pasos seguidos durante el desarrollo de este proyecto

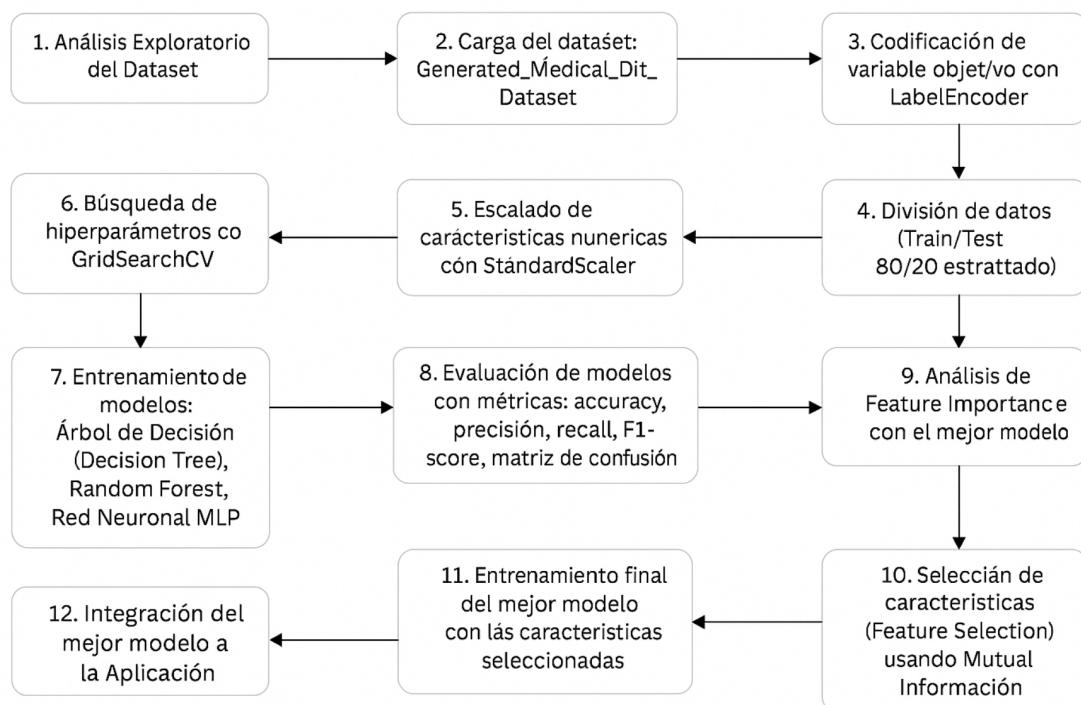


Figura 1: Pipeline del Proyecto

1. Análisis Exploratorio.

El análisis exploratorio constituye una etapa esencial dentro del ciclo de desarrollo de modelos de machine learning, ya que permite entender la estructura del dataset, detectar valores atípicos, identificar problemas de calidad de datos (como valores nulos o codificaciones incorrectas) y tomar decisiones informadas sobre el preprocesamiento. En este proyecto, se empleó un dataset personalizado denominado "Generated_Medical_Diet_Dataset.csv", el cual se construyó a partir de investigaciones académicas sobre recomendaciones dietéticas para condiciones crónicas sintetizados por Elicit.

¹El dataset fue construido a partir de una síntesis de evidencia proveniente de estudios de alto impacto, incluyendo revisiones sistemáticas, metaanálisis y ensayos clínicos controlados aleatorizados. Entre estos estudios se destacan:

- *Barnett et al.*, "A Systematic Review and Meta-Analysis of Controlled Trials on Dietary Interventions via Digital Health Platforms in Chronic Conditions", 2022. [5]
- *Chauveau et al.*, "Mediterranean Diet Interventions in Patients with Chronic Kidney Disease: A Review Article", 2018. [6]
- *Anderson et al.*, "Systematic Review and Meta-Analysis of High-Fiber, High-Carbohydrate, and Low-Glycemic Index Diets in Diabetes Management", 2004. [7]
- *Appel et al.*, "Crossover Feeding Study on Carbohydrate-, Protein-, and Unsaturated Fat–Rich Diets in Adults with Prehypertension or Stage 1 Hypertension", 2005. [8]
- *Elmer et al.*, "Randomized Controlled Trial on the DASH Diet in Adults with Prehypertension or Stage 1 Hypertension", 2006. [9]
- *McMillan-Price et al.*, "RCT Comparing High-Carb and High-Protein Diets (High vs. Low GI) in Overweight or Obese Young Adults", 2006. [10]
- *Yancy et al.*, "Low-Carbohydrate Ketogenic Diet vs. Low-Fat Diet in Obese Adults with Hyperlipidemia: A Randomized Trial", 2004. [11]
- *Esposito et al.*, "Mediterranean-Style vs. Low-Fat Diet in Overweight Patients with Newly Diagnosed Type 2 Diabetes: An RCT", 2009. [12]

¹ Elicit es una herramienta de inteligencia artificial desarrollada por Ought.org que sintetiza evidencia científica a partir de múltiples fuentes académicas.

Elicit, *Dietary Recommendations for Chronic Conditions*. [Online].

Available: <https://elicit.com/review/6ab9b8e9-3fea-452f-b7ef-60e220db005b>

Elicit, *Dietary Strategies for Comorbid Chronic Conditions*. [Online].

Available: <https://elicit.com/review/e8c47975-eb80-4f2d-90f6-429ad36f931f>

- *Ajala et al.*, "Systematic Review and Meta-Analysis of Dietary Approaches for Type 2 Diabetes: Low-Carb, Vegetarian, Vegan, Low-GI, High-Fiber, Mediterranean, and High-Protein", 2013. [13]
- *Campbell et al.*, "RCT on Self-Managed Dietary Sodium Restriction in Adults with Hypertension and Chronic Kidney Disease", 2017. [14]
- *Yoon et al.*, "Prospective Cohort Study on Sodium Intake in Populations With and Without Hypertension", 2017. [15]
- *Zaharieva and Riddell*, "Literature Review of Low Glycemic Index Diets in Type 1 and Type 2 Diabetes Across 29 Trials", 2023. [16]

Este dataset está compuesto por 1401 registros y 13 columnas (12 predictoras + 1 variable objetivo). La variable objetivo corresponde a Recommended_Meal_Plan_Medical, que representa la dieta recomendada para cada individuo, mientras que las variables predictoras incluyen factores tales como

- Age
- Activity_Level
- Sodium_mg
- Sugar_g
- Daily_Calories
- Severity
- Diabetes
- Obesity
- Heart Disease
- Hypertension
- Kidney_Disease
- No_Condición

Esta base de conocimiento fue sintetizada para establecer una correspondencia directa entre condiciones clínicas y patrones alimentarios recomendados, permitiendo una correcta asignación de clases en la variable objetivo Recommended_Meal_Plan_Medical.

AGE	ACTIVITY_LEVEL	SODIUM_MG	SUGAR_G	DAILY_CALORIES_TARGET	SEVERITY	DIABETES	OBESEITY	HEART_DISEASE	HYPERTENSION	KIDNEY_DISEASE	NO_CONDITION	RECOMMENDED_MEAL_PLAN_MEDICAL
34	3	1606	22	2355	0	0	0	0	0	0	1	Balanced
39	2	1889	32	1970	0	0	0	0	0	0	1	Balanced
54	2	2206	30	2102	0	0	0	0	0	0	1	Balanced
31	3	1907	33	2144	0	0	0	0	0	0	1	Balanced
61	2	720	19	2039	2	0	0	1	1	0	0	DASH
45	3	590	23	1697	2	0	0	1	1	0	0	DASH
50	2	656	24	1765	2	0	0	1	1	0	0	DASH
37	5	2012	5	1818	1	1	0	0	0	0	0	Low-Carb
46	4	1886	3	2155	1	1	0	0	0	0	0	Low-Carb
35	3	2073	0	1830	1	1	0	0	0	0	0	Low-Carb
46	3	2140	18	2164	1	0	1	0	0	0	0	Mediterranean

Tabla 1: Resumen de la disposición del Dataset

Durante la etapa de limpieza y preprocesamiento, se realizaron los siguientes pasos:

1. Cargar el Dataset Generated_Medical_Diet_Dataset.csv
2. Se reviso la distribucion de las clases de la variable objetivo y se determine que era optima por lo que no requeria de un balanceo ver Figura 2.
3. Se codificó la variable Recommended_Meal_Plan_Medical a valores numéricos con LabelEncoder.
4. Se realizó un Ranking de Características inicial con Random Forest, con el fin de observar las relaciones de las variables predictivas y la variable objetivo (ver Figura 3). Este ranking se obtuvo evaluando la reducción promedio de impureza (índice Gini) que cada variable aporta a lo largo de los árboles del modelo.
5. Se realizó una visualización del espacio latente mediante t-SNE, con el objetivo de explorar la separabilidad de clases en función de las variables predictoras ver Figura 11.
6. Se separaron las variables predictoras (X) de la variable objetivo (y):
7. Para el modelo MLP se aplicó escalamiento estándar (StandardScaler) a las variables numéricas.
8. Luego del preprocesamiento, el dataset se divide en un 80% para entrenamiento y 20% para prueba

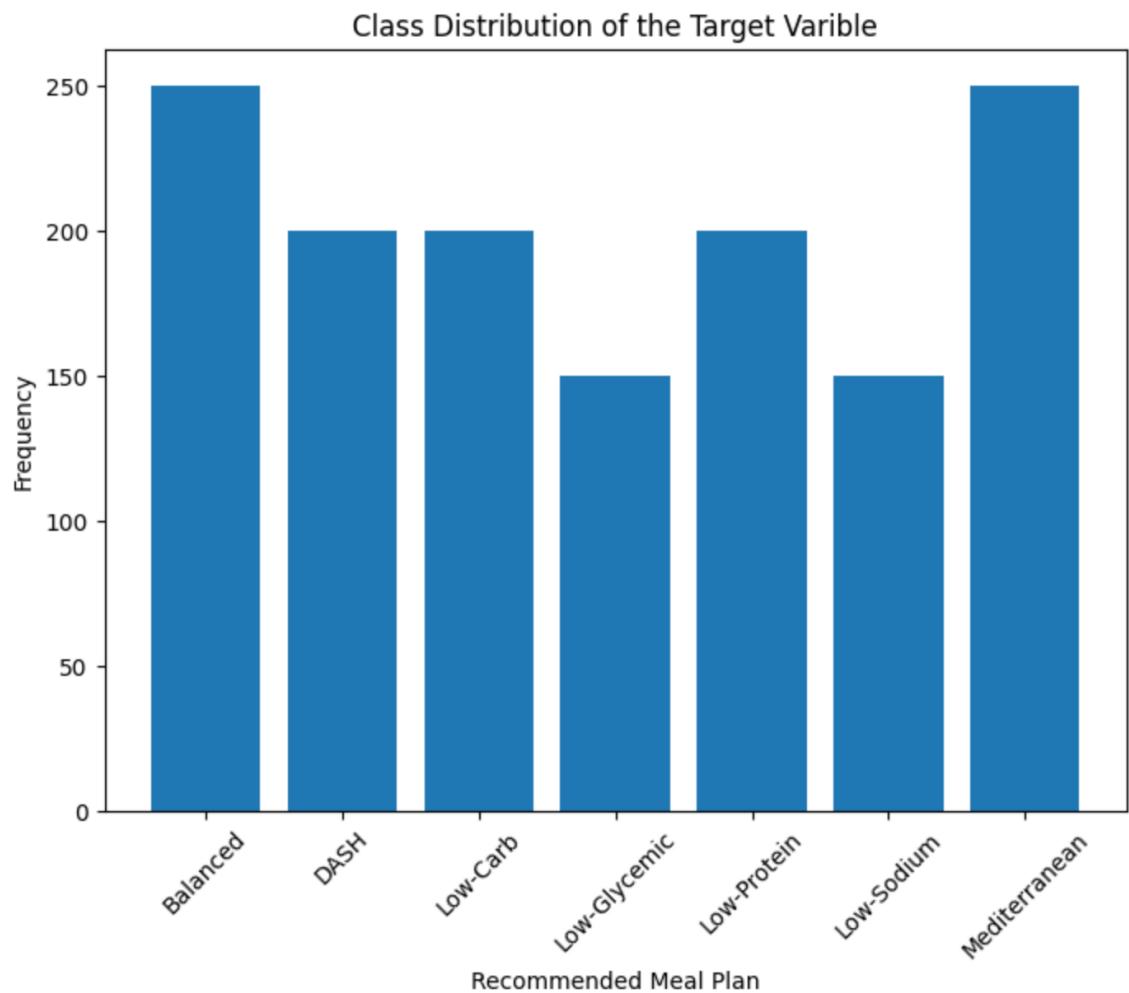


Figura 2: Distribución variable objetivo

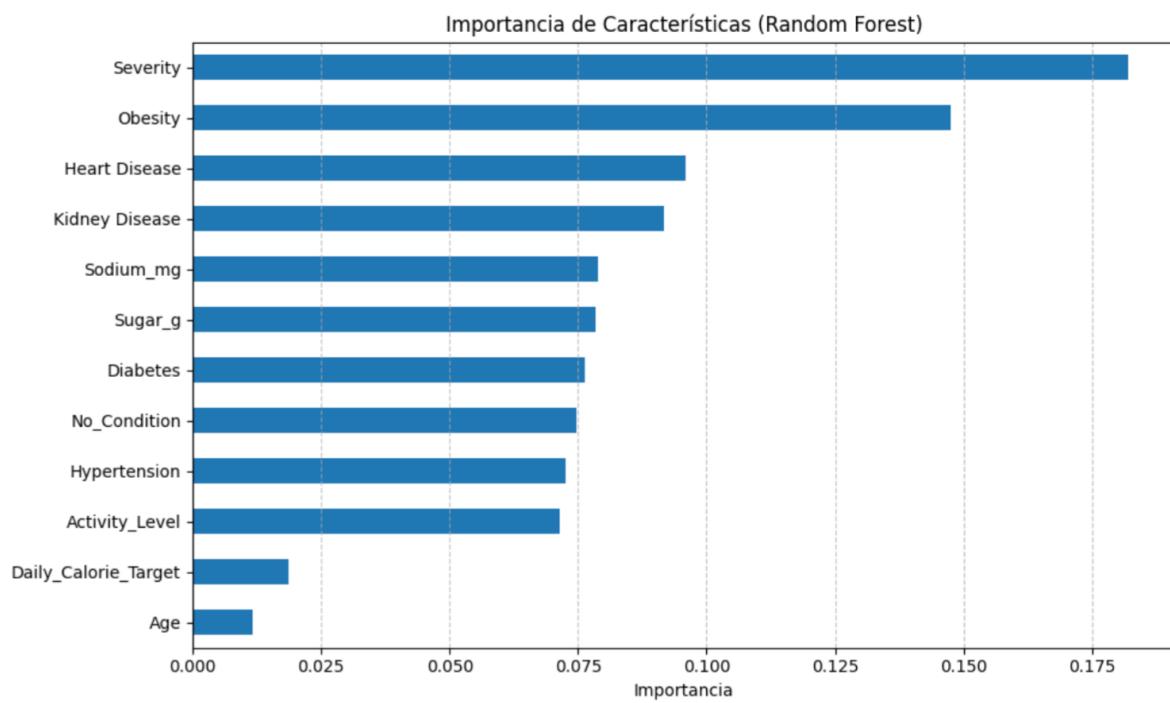


Figura 3: Ranking de Características RF

2. Definición de métricas de rendimiento y selección del Baseline

Una vez realizado el análisis exploratorio del Dataset, podemos proceder con el siguiente paso. Definir las métricas de rendimiento, dado que es un problema de clasificación multiclas, las métricas a utilizar serán:

Precisión: Nos da la proporción de predicciones correctas de cada modelo.

Matriz de confusión: Permite visualizar errores de clasificación de manera sencilla.

Antes de detallar las métricas seleccionadas, es importante comprender dos indicadores clave:

- Recall: también conocido como *sensibilidad*, mide la proporción de verdaderos positivos entre todos los casos reales de una clase. Es útil para evaluar cuántos casos positivos reales fueron correctamente identificados por el modelo.

- F1-Score: es la media armónica entre precisión y recall. Esta métrica resulta especialmente útil cuando existe un desbalance entre clases, ya que penaliza tanto los falsos positivos como los falsos negativos.

Por tanto, las métricas de rendimiento utilizadas serán:

- Precisión (Accuracy): indica la proporción total de predicciones correctas realizadas por el modelo.
- Matriz de confusión: permite visualizar de forma detallada los aciertos y errores de clasificación por clase.
- Recall y F1-Score: ofrecen una evaluación más profunda del rendimiento, especialmente útil en contextos donde los errores de clasificación pueden tener diferente impacto.

Para el proceso de clasificación y selección de dietas, se utilizarán Decision Trees, Random Forest y Redes Neuronales MLP, ya que estas técnicas han demostrado ser eficaces en tareas de clasificación y permiten establecer una línea base de rendimiento para contrastar con modelos más complejos. Según Tito et al. [17], los árboles de decisión han sido empleados exitosamente en la predicción de la deserción estudiantil, mostrando resultados significativos por su capacidad de manejar relaciones no lineales, interpretar reglas de decisión y ofrecer modelos fácilmente interpretables. En este estudio, se utilizará Decision Trees como modelo baseline, debido a su simplicidad, rápida ejecución y facilidad de interpretación, permitiendo comparar su desempeño con modelos más robustos como Random Forest y MLP.

Decision Tree:

El modelo Decision Tree fue configurado con el objetivo de controlar el sobreajuste, restringiendo la complejidad del árbol mediante una búsqueda en malla (*GridSearchCV*) con validación cruzada tipo K-Fold (5 Folds). Los mejores hiperparámetros encontrados fueron: `max_depth=10`, `min_samples_split=2`, y `min_samples_leaf=1` (ver Tabla 2). Estos

parámetros permitieron al árbol alcanzar un equilibrio entre profundidad y generalización, evitando que memorice el conjunto de entrenamiento.

PARÁMETRO	VALOR
MAX_DEPTH	10
MIN_SAMPLES_LEAF	1
MIN_SAMPLES_SPLIT	2

Tabla 2: Mejores Parametros Decision Tree

Los resultados obtenidos reflejan un rendimiento sólido, alcanzando un accuracy del 98% sobre el conjunto de prueba. El análisis clase por clase (ver Tabla 3) muestra que el modelo se comporta consistentemente bien en casi todas las clases. Aunque clases como “Low-Carb” y “Low-Glycemic” presentaron un desempeño ligeramente menor, este aun esta dentro de márgenes aceptables para tareas de clasificación multiclas.

CLASE	PRECISIÓN	RECALL	F1-SCORE	SOPORTE (#INSTANCIAS)
BALANCED	1.00	1.00	1.00	50
DASH	1.00	1.00	1.00	40
LOW-CARB	0.87	1.00	0.93	40
LOW-GLYCEMIC	1.00	0.80	0.89	30
LOW-PROTEIN	1.00	1.00	1.00	40
LOW-SODIUM	1.00	1.00	1.00	30
MEDITERRANEAN	1.00	1.00	1.00	50
ACCURACY			0.98	280
MACRO AVG	0.98	0.97	0.97	280
WEIGHTED AVG	0.98	0.98	0.98	280

Tabla 3: Reporte de Clasificación Decision Tree

Este comportamiento sugiere que el modelo logra capturar patrones relevantes sin caer en sobreajuste, lo cual también puede observarse en la matriz de confusión (ver Figura 4), donde los errores están distribuidos de manera razonable de acuerdo con lo visto en el Reporte de Clasificación.

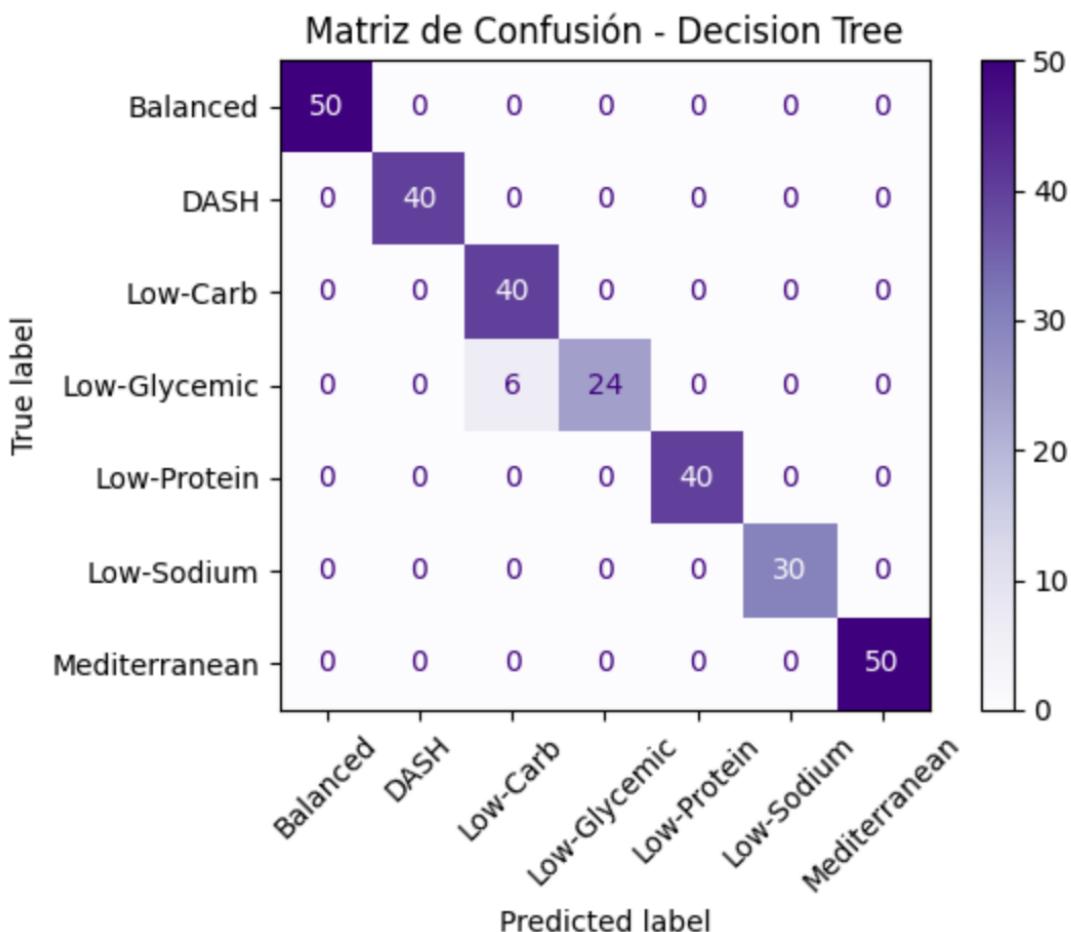


Figura 4: Matriz de Confusión DT

Random Forest:

El modelo Random Forest fue configurado utilizando una búsqueda en malla (GridSearchCV) sobre un conjunto extenso de hiperparámetros, evaluando combinaciones mediante validación cruzada tipo K-Fold (5 folds). Los hiperparámetros explorados incluyeron: número de árboles (n_estimators), profundidad máxima (max_depth), número mínimo de muestras por nodo (min_samples_split, min_samples_leaf), el método de selección de características (max_features) y si se aplica bootstrap (bootstrap). La mejor configuración encontrada se detalla en la Tabla 4.

PARÁMETRO	VALOR
N_ESTIMATORS	100
MAX_DEPTH	15
MIN_SAMPLES_SPLIT	20
MIN_SAMPLES_LEAF	2
MAX_FEATURES	sqrt
BOOTSTRAP	false

Tabla 4: Mejores Parámetros - Random Forest

Con esta configuración, el modelo alcanzó un accuracy del 98% en el conjunto de prueba. La evaluación clase por clase (ver Tabla 5) muestra un excelente desempeño en todas las clases, que alcanzaron un F1-score cercano o igual a 1.00 con un desempeño ligeramente mejor en Low-Carb y Low-Glycemic en comparación al Decission Tree .

CLASE	PRECISIÓN	RECALL	F1-SCORE	SOPORTE (#INSTANCIAS)
BALANCED	1.00	1.00	1.00	50
DASH	1.00	1.00	1.00	40
LOW-CARB	0.89	1.00	0.94	40
LOW-GLYCEMIC	1.00	0.83	0.91	30
LOW-PROTEIN	1.00	1.00	1.00	40
LOW-SODIUM	1.00	1.00	1.00	30
MEDITERRANEAN	1.00	1.00	1.00	50
ACCURACY			0.98	280
MACRO AVG	0.98	0.98	0.98	280
WEIGHTED AVG	0.98	0.98	0.98	280

Tabla 5: Reporte de Clasificación - Random Forest

La Figura 5 (matriz de confusión) muestra el buen rendimiento, con muy pocas instancias mal clasificadas. El modelo destaca por su capacidad para mantener una alta precisión y recall sin caer en sobreajuste, en parte gracias a la aleatorización de muestras y características propias del algoritmo Random Forest.

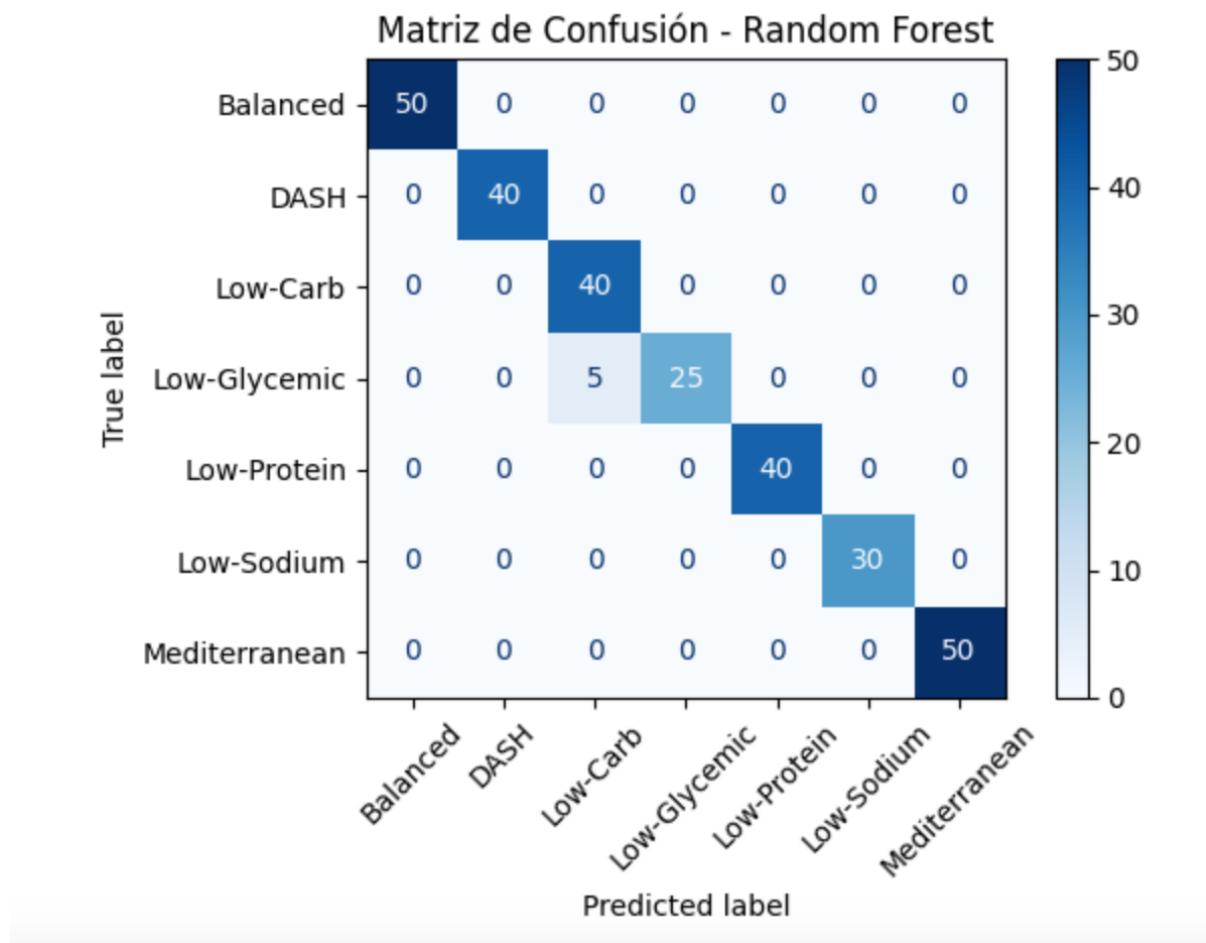


Figura 5: Matriz de Confusión RF

Además, al comparar las curvas de accuracy de validación y entrenamiento (ver Figura 6), se observa que no existe una brecha considerable entre ambos, lo que valida su robusta capacidad de generalización. El F1 macro score considera el desempeño en todas las clases por igual, lo que resulta especialmente útil en problemas de clasificación multiclas. Por otro lado el eje horizontal representa el índice de configuración generado durante la búsqueda en malla (Grid Search, es decir cada punto corresponde a una configuración específica de hiperparámetros del modelo Random Forest.

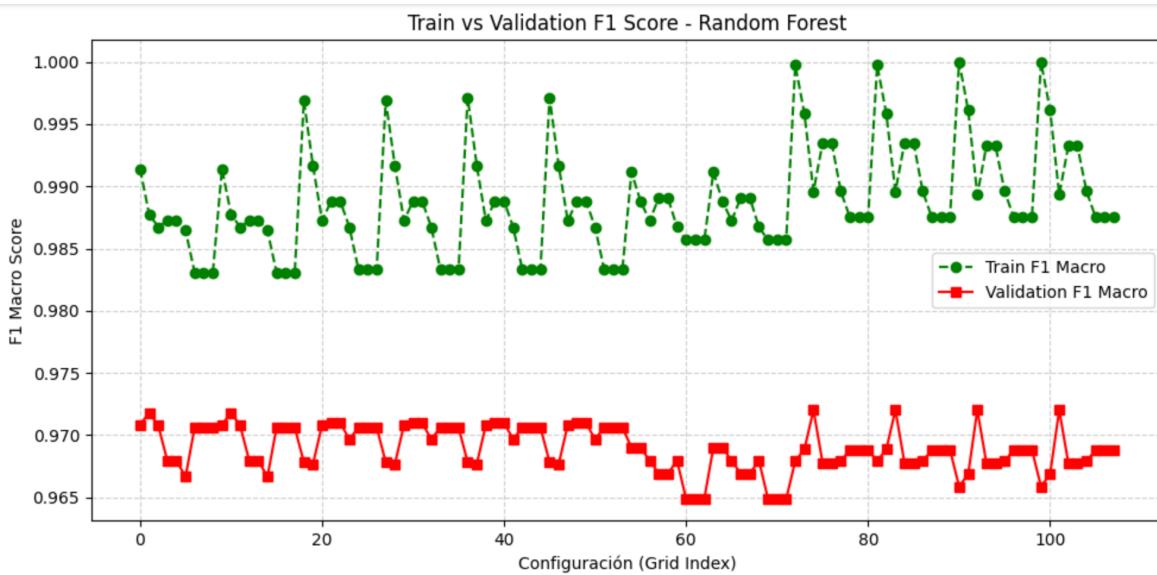


Figura 6: Curvas de validación RF

Si bien el modelo Random Forest presentó un excelente desempeño y evidenció una alta capacidad de generalización, se consideró oportuno evaluar un enfoque de tipo red neuronal, específicamente un Multilayer Perceptron (MLP), con el objetivo de explorar modelos capaces de capturar relaciones no lineales más complejas entre las características y la variable objetivo. Las redes neuronales, a diferencia de los modelos basados en árboles, tienen la capacidad de ajustar funciones altamente no lineales, lo que puede ser beneficioso en problemas de clasificación multiclas como en el presente trabajo. Además, su flexibilidad y adaptabilidad las convierten en una alternativa interesante, especialmente cuando se combinan con técnicas de regularización que permiten evitar el sobreajuste, como la penalización L2, el *dropout* y la normalización por lotes (*batch normalization*), tal como lo implementaron Singh et al. en su estudio sobre regularización en redes neuronales profundas aplicadas a imágenes denoised [18].

Redes Neuronales:

El modelo MLP fue diseñado como una alternativa más flexible a los modelos basados en árboles, con el objetivo de intentar capturar relaciones no lineales presentes en el conjunto de datos. Para ello, se construyó una red neuronal con dos capas ocultas densas: la primera con 128 neuronas y la segunda con 64, ambas implementadas con regularización L2 (coeficiente de 0.001), Batch Normalization y Dropout (0.5 y 0.4 respectivamente) para reducir el

sobreajuste. Adicionalmente, se agregó ruido gaussiano al input para robustecer el aprendizaje y mejorar la generalización. Aunque en este estudio se utilizó la regularización L2, también se consideró la técnica L1 como alternativa. La regularización L1 tiene la particularidad de inducir sparsity, eliminando pesos irrelevantes y promoviendo modelos más simples, mientras que L2 tiende a distribuir la penalización uniformemente. En problemas donde es necesario seleccionar un subconjunto reducido de características, L1 puede ofrecer ventajas. En nuestro caso, se optó por L2 debido a su estabilidad durante el entrenamiento y su buena sinergia con el optimizador Adam.

Es crucial diferenciar entre los conceptos de normalización y regularización dentro del contexto de redes neuronales. La *normalización*, como la aplicada mediante *Batch Normalization*, tiene como objetivo estabilizar el entrenamiento ajustando la distribución de las activaciones, lo que permite una convergencia más rápida y precisa. Por otro lado, la *regularización* se refiere a técnicas diseñadas para evitar el sobreajuste del modelo, tales como Dropout o penalizaciones L1 y L2. Mientras que la normalización actúa sobre las activaciones internas para mantenerlas dentro de rangos adecuados, la regularización impone restricciones directas sobre los parámetros del modelo, promoviendo así una mejor generalización.

El modelo fue entrenado durante un máximo de 100 épocas, con early stopping si la pérdida de validación no mejoraba en 7 épocas consecutivas. Se utilizó el optimizador Adam con una tasa de aprendizaje inicial de 0.0001 y escalado de características mediante StandardScaler. La división de datos se realizó en un 80/20 para entrenamiento y prueba, estratificando por clase para conservar la distribución original.

Los resultados muestran que MLP logra un rendimiento excelente (accuracy de 99%), con una presición perfecta en la mayoría de clases, este modelo presenta una mejora al clasificar las clases Low-Carb y Low-Glysemic , y se mantiene en la linea de los observado en los modelos anteriores teniendo mas dificultades a la hora de distinguir entre estas dos clases (ver Tabla 6).

CLASE	PRECISIÓN	RECALL	F1-SCORE	SOPORTE
BALANCED	1.00	1.00	1.00	50
DASH	1.00	1.00	1.00	40
LOW-CARB	0.97	0.97	0.97	40
LOW-GLYCEMIC	0.97	0.97	0.97	30
LOW-PROTEIN	1.00	1.00	1.00	40
LOW-SODIUM	1.00	1.00	1.00	30
MEDITERRANEAN	1.00	1.00	1.00	50
ACCURACY			0.99	280
MACRO AVG	0.99	0.99	0.99	280
WEIGHTED AVG	0.99	0.99	0.99	280

Tabla 6: Reporte de Clasificación – MLP

Si observamos la Figura 7, podremos observar una disminución progresiva y estable de la pérdida, tanto en el conjunto de entrenamiento como en el de validación. Esto sugiere que no hay señales evidentes de sobreajuste. La cercanía entre ambas curvas sugiere una buena capacidad de generalización, lo cual se alinea con las métricas obtenidas en la evaluación final. En términos de desempeño, el MLP alcanza un accuracy del 99%, superando a los modelos Random Forest y Decision Tree, posicionándose como una alternativa robusta dentro de los modelos evaluados.

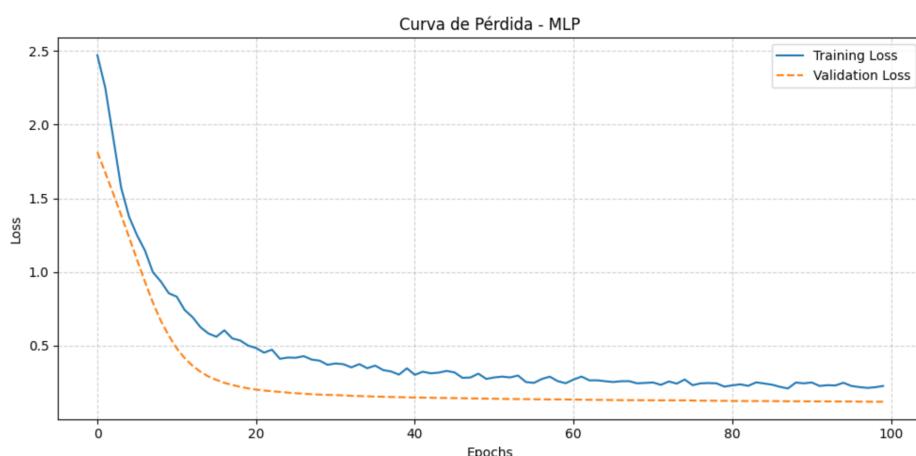


Figura 7: Graficas loss Neural Network

Luego de la implementación de los distintos algoritmos y el análisis de su desempeño en la predicción de dietas, se ha decidido utilizar el modelo MLP como opción final. Si bien modelos como Random Forest demostraron un rendimiento casi equivalente en términos de precisión y capacidad de generalización, MLP fue seleccionado principalmente por su comportamiento más estable durante el entrenamiento y validación, así como por su alto rendimiento en todas las clases evaluadas. Esto lo convierte en una alternativa robusta y práctica para el entorno móvil planteado, garantizando tanto eficiencia como confiabilidad en la aplicación final.

3. Análisis de Reducción de Dimensionalidad.

Con el objetivo de optimizar la eficiencia del modelo y facilitar su implementación práctica, se realizó un análisis para reducir la cantidad de características requeridas. Para ello, se empleó una técnica de selección de atributos basada en la información mutua, con el fin de identificar y priorizar aquellas variables que aportan mayor valor a la predicción. Esta estrategia busca construir una aplicación más accesible y eficiente, minimizando la carga cognitiva y operativa del usuario. En contextos como la salud y la nutrición, es fundamental que la interacción sea rápida y sencilla, evitando la necesidad de ingresar un gran volumen de datos. La reducción de características también ayuda a eliminar información redundante o irrelevante, lo cual contribuye a mejorar la precisión del modelo al reducir el ruido y la complejidad innecesaria. Este análisis, por tanto, apunta a encontrar un equilibrio entre rendimiento predictivo y facilidad de uso, aspectos clave para el éxito de una aplicación para dispositivos móviles.

Feature selection

Para la selección de características se implementó un enfoque basado en la métrica de información mutua utilizando el método `mutual_info_classif` de la librería scikit-learn, como se puede observar en la Figura 8. La utilización de este tipo de técnicas ha demostrado ser efectiva en aplicaciones reales, tal como lo evidencian Santos, Carrasco y Martínez en [19], quienes aplicaron la reducción de dimensionalidad para la estimación de parámetros estelares, logrando mejorar tanto la precisión como la interpretabilidad de los modelos.

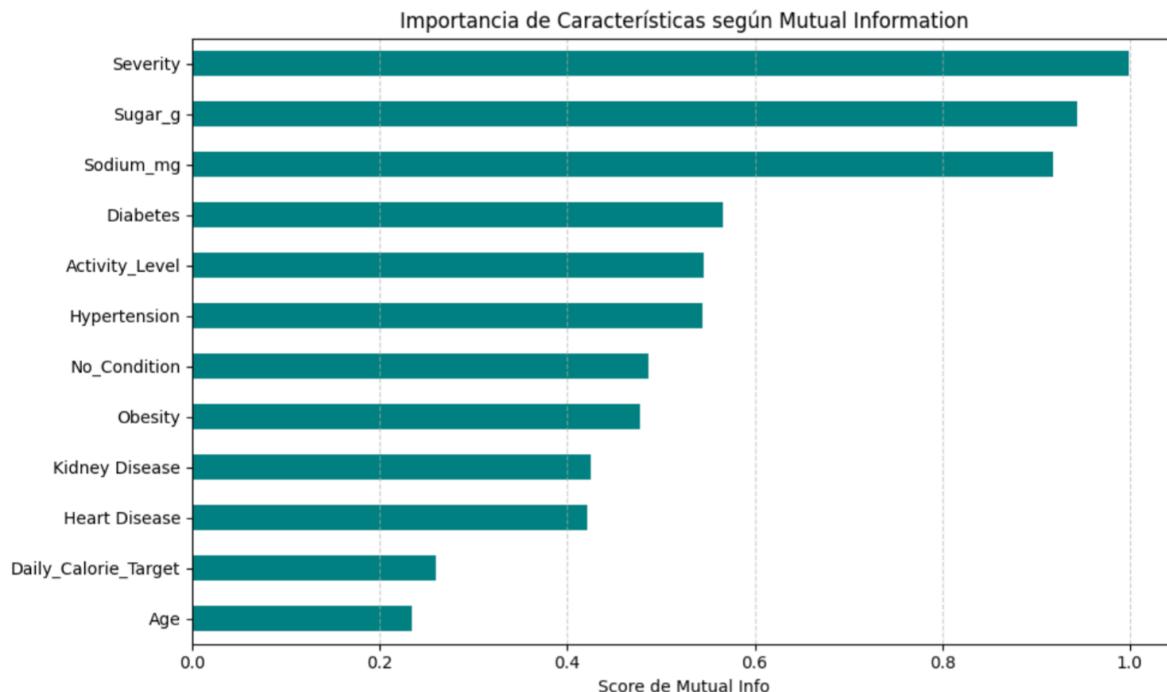


Figura 8: Características mutual_info

Con base en estos resultados, se seleccionaron las diez características más informativas según la métrica de información mutua. Estas variables fueron utilizadas como entrada del modelo final, garantizando un balance entre capacidad predictiva y simplicidad. Esta selección permite optimizar el modelo sin comprometer su rendimiento, a la vez que mejora la experiencia del usuario al reducir la cantidad de datos requeridos para generar una recomendación. Las características seleccionadas se pueden observar en la Tabla 7.

El análisis de importancia de características obtenido mediante Random Forest, ilustrado en la Figura 9, mostró una jerarquía consistente con los resultados obtenidos por información mutua. En ambas visualizaciones, la variable Severity destaca como la más relevante, seguida de otras condiciones clínicas clave como Obesity, Heart Disease y Kidney Disease.

Aunque el orden de importancia varía ligeramente entre ambos métodos, las diez características seleccionadas resultaron ser las mismas. Esto se debe a que la información mutua mide la dependencia estadística entre cada característica y la variable objetivo, mientras que el Ranking de Random Forest evalúa la contribución de cada característica a la reducción

de impureza en los arboles de decision. Esta convergencia entre métodos refuerza la validez de las variables seleccionadas y justifica su uso como características principales para el sistema de recomendación.

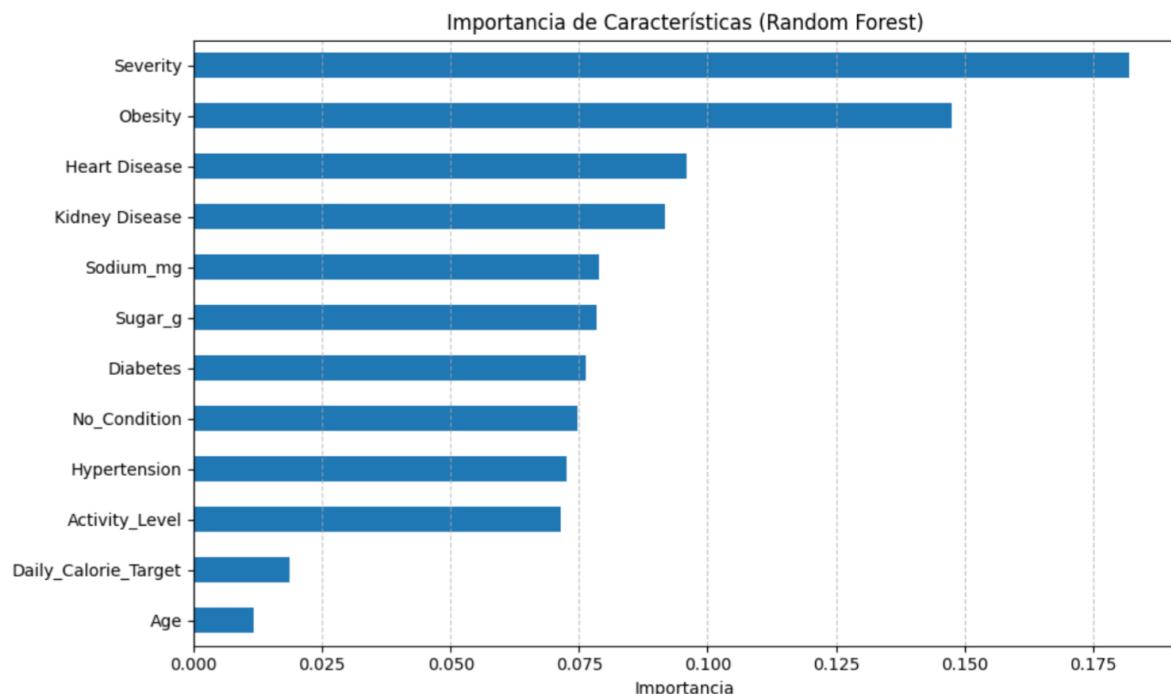


Figura 9: Características Random Forest

Nº	CARACTERÍSTICA
1	Activity_level
2	Sodium_mg
3	Sugar_g
4	Severity
5	Diabetes
6	Obesity
7	Heart Disease
8	Hypertension
9	Kidney Disease
10	No_Condition

Tabla 7: Selección de las 10 mejores características

Posteriormente, se entrenó un modelo Multi-Layer Perceptron (MLP) sobre el conjunto de datos reducido tras la aplicación de técnicas de selección de características. La arquitectura del modelo incluyó una capa de entrada acorde al número de variables seleccionadas, seguida por dos capas ocultas densas con 128 y 64 neuronas respectivamente, activación ReLU, regularización L2 (0.001), normalización por lotes (*Batch Normalization*) y funciones de abandono (*Dropout*) del 50% y 40%. Para la capa de salida se utilizó una activación *softmax* con tantas neuronas como clases en el conjunto objetivo. Esta configuración permitió obtener un modelo altamente preciso y generalizable, con curvas de pérdida que reflejaron un entrenamiento estable y sin sobreajuste.

La selección de características permitió simplificar el proceso de aprendizaje al solicitar la cantidad de entradas requeridas por el modelo, obteniendo al mismo tiempo un modelo robusto. A pesar de la reducción de información, el modelo logró mantener una alta precisión del 99% en el conjunto de prueba, demostrando que es factible simplificar la cantidad de características sin afectar significativamente el rendimiento del modelo. Este modelo resulta especialmente útil para la aplicación, ya que al solicitar menos variables facilita la recolección de datos y mejora la experiencia de usuario.

Así mismo, el reporte de clasificación, podemos muestra un excelente desempeño para todas las clases; esto se ve reforzado por lo visto en la tabla 8 y la Figura 10, siendo superado muy ligeramente por el MLP sin feature selection, ya que este generaliza algo mejor las clases Low-Carb y Low Glycemyc.

CLASE	PRECISIÓN	RECALL	F1-SCORE	SOPORTE
BALANCED	1.00	1.00	1.00	50
DASH	1.00	1.00	1.00	50
LOW-CARB	0.95	0.97	0.89	50
LOW-GLYCEMIC	0.97	0.93	0.91	50
LOW-PROTEIN	1.00	1.00	1.00	50
LOW-SODIUM	1.00	1.00	1.00	50
MEDITERRANEAN	1.00	1.00	1.00	50
ACCURACY			0.99	350
MACRO AVG	0.99	0.99	0.99	350
WEIGHTED AVG	0.99	0.99	0.99	350

Tabla 8: Reporte de Clasificación del modelo MLP con FS

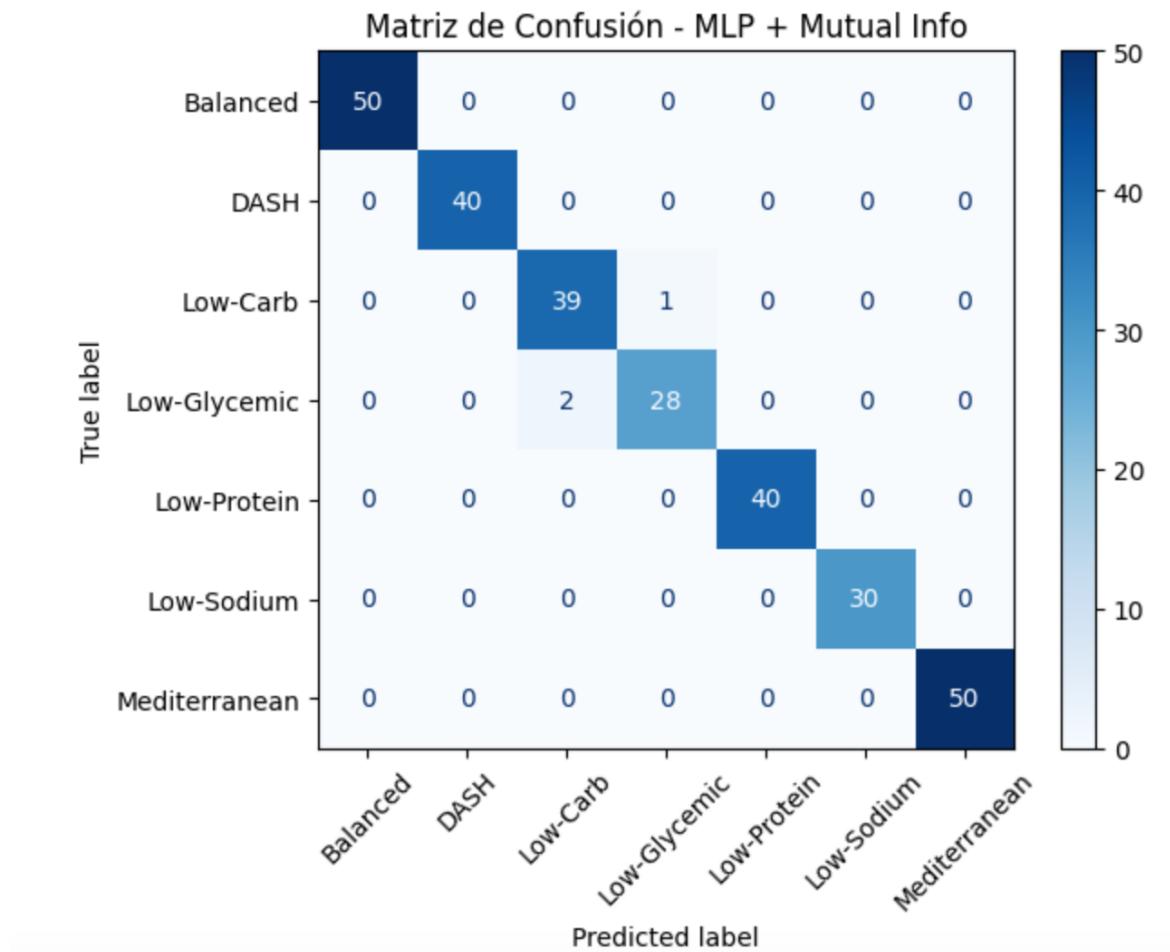


Figura 10: Matriz de Confusión MLP-FS

Finalmente, como se evidencia en la Figura 11, el análisis de la estructura interna del dataset mediante la técnica de reducción de dimensionalidad PCA respalda visualmente el buen desempeño observado en las métricas de clasificación en todos los modelos. PCA permite reducir la dimensionalidad del conjunto original proyectándolo en dos componentes principales que conservan la mayor varianza posible. El resultado es un espacio latente es decir, una representación comprimida pero significativa de los datos, donde se observa una clara separación entre las clases, con agrupaciones bien definidas y mínimas zonas de solapamiento. Esta organización sugiere que las variables seleccionadas capturan eficazmente las diferencias entre tipos de dieta, facilitando que los modelos como MLP aprendan de manera

precisa incluso con un número limitado de características. Así pues, esta validación visual justifica las métricas obtenidas en los distintos modelos, y refuerza la elección del modelo MLP con feature selection como la mejor opción para integrar en el sistema de recomendación. A continuación, se presenta el proceso de desarrollo e implementación de la aplicación móvil, así como su integración con el modelo entrenado.

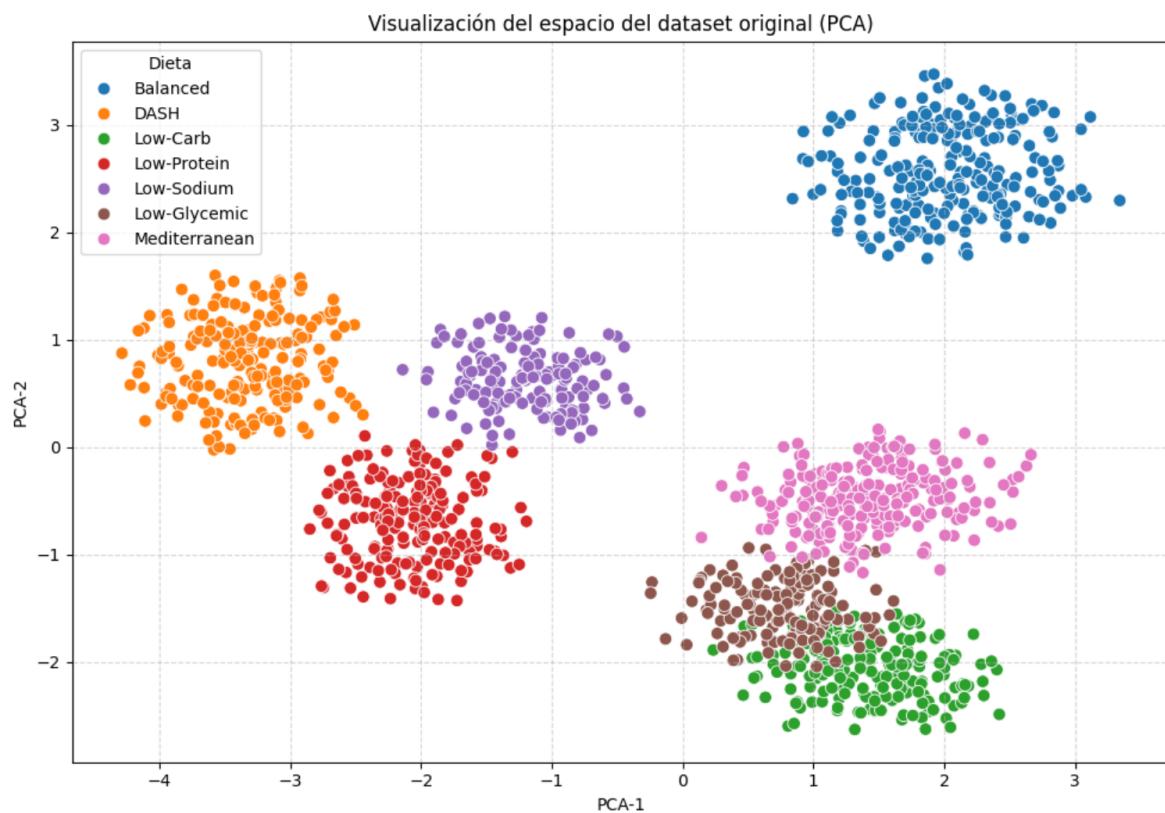


Figura 11: Visualización espacio con PCA

3. Desarrollo de la app Móvil

Para la implementación de la solución propuesta, se desarrolló una aplicación móvil denominada "Predicción de Dietas con IA" utilizando el framework Flutter. La elección de Flutter se basó en su capacidad para crear aplicaciones multiplataforma, permitiendo que una misma base de código sea desplegada tanto en sistemas operativos Android como iOS. Esta característica es fundamental en el contexto del proyecto, ya que facilita su adopción por parte de un público amplio sin necesidad de desarrollar y mantener dos aplicaciones independientes, optimizando así recursos de desarrollo y mantenimiento. Además, Flutter ha sido utilizado con

éxito en proyectos similares de predicción de enfermedades mediante inteligencia artificial, como se presenta en [20], donde se implementó una aplicación de predicción basada en síntomas con modelos de aprendizaje automático, validando su eficacia para aplicaciones médicas.

Integración del modelo mediante el uso de Flask como backend:

El proceso de predicción se lleva a cabo mediante la interacción entre la aplicación Flutter y una API desarrollada en Flask. La elección de Flask se justifica por su naturaleza ligera, modular y ampliamente utilizada en la construcción de servicios RESTful para modelos de aprendizaje automático. Este framework es especialmente adecuado para proyectos que requieren crear APIs rápidas y eficientes, integrando modelos de machine learning en entornos productivos. Su adopción ha sido documentada en diversas aplicaciones similares, como en el trabajo de Patil et al. [21], donde Flask fue empleado exitosamente para la implementación de un sistema de reconocimiento facial basado en redes neuronales, demostrando su eficacia para el desarrollo de APIs destinadas a tareas de inferencia en tiempo real y su facilidad de integración con modelos desarrollados en Python.²

Arquitectura de la Aplicación:

La arquitectura de la aplicación se diseñó bajo un enfoque cliente-servidor, priorizando la modularidad, la escalabilidad y la facilidad de mantenimiento. Como se observa en la Figura 12, la aplicación desarrollada en Flutter actúa como cliente para acceder al servicio de recomendación de dietas, recolectando los datos necesarios para la recomendación y enviándolos a través de una solicitud HTTP a la API implementada en Flask. Esta API se encarga de procesar los datos, aplicar el modelo de aprendizaje automático MLP con Feature Selection y devolver una recomendación dietética personalizada. Actualmente, en esta primera

² Google Developers, “Flutter,” *flutter.dev*. [Online]. Available: <https://flutter.dev>.

P. Ronacher, “Welcome to Flask,” *Flask Documentation*, Pallets Projects. [Online]. Available: <https://flask.palletsprojects.com>

versión prototipo, el servidor Flask se ejecuta de forma local a través de la terminal en sistemas macOS o mediante la consola en equipos con Windows. Esta configuración permite validar la lógica de funcionamiento e integración entre los distintos componentes de la aplicación antes de su despliegue definitivo en un entorno en la nube. Esta estructura permite desacoplar la lógica de presentación de la lógica de predicción, lo que facilita futuras actualizaciones del modelo sin afectar la interfaz de usuario. Además, la elección de herramientas ligeras y ampliamente adoptadas como Flutter y Flask garantiza un flujo eficiente de datos, así como una integración fluida entre los componentes de la solución.

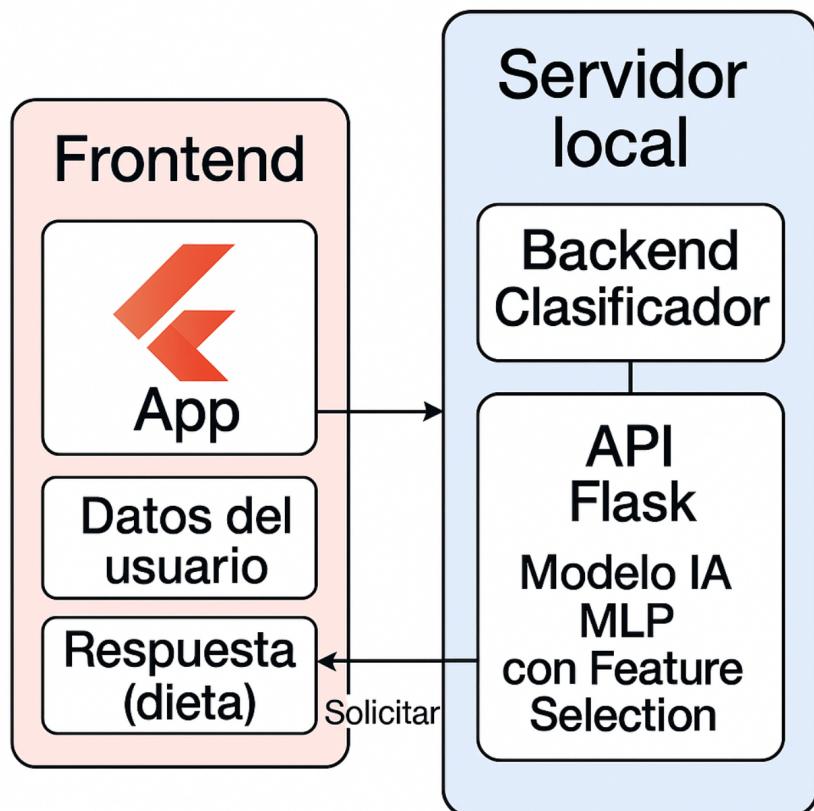


Figura 12: Arquitectura de la Aplicación

El diagrama de secuencia representado en la Figura 13 muestra el flujo de comunicación entre los distintos componentes que conforman la aplicación de recomendación de dietas, la cual

integra una interfaz desarrollada en Flutter, una API basada en Flask y un modelo de inteligencia artificial. El proceso se inicia cuando el usuario ingresa sus datos clínicos y de estilo de vida en la app, los cuales son enviados a través de una solicitud HTTP POST al endpoint /api/recomendar en un JSON, ver Figura 14 y la Tabla 9. La API recibe estos datos, los procesa y los escala utilizando un StandardScaler. Posteriormente, se realiza la predicción mediante el modelo de *machine learning* (MLP) que ha sido previamente entrenado con técnicas de selección de características. Una vez determinada la dieta más adecuada, la API accede directamente a un archivo estructurado en formato JSON (menus_por_dieta.json), el cual contiene menús predefinidos y validados para cada tipo de dieta, organizados por día y divididos en desayuno, comida y cena. De esta manera, tanto la dieta como el menú correspondiente son devueltos en una única respuesta, permitiendo entregar al usuario una experiencia inmediata, sin necesidad de realizar solicitudes adicionales ni generar menús dinámicamente en tiempo real, optimizando así la eficiencia del sistema. Finalmente, el menú diario es mostrado en la aplicación, brindando una guía práctica y adaptada al perfil de salud del usuario.

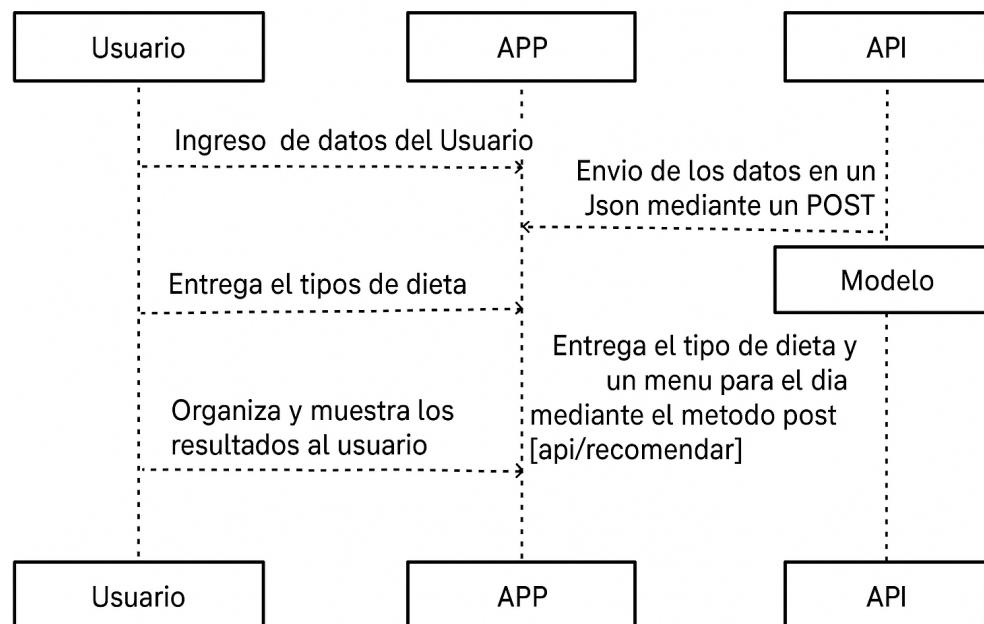


Figura 13: Diagrama de flujo de la APP

URI	METODO	API
/API/RECOMENDAR	POST	<p>Recive los datos del cliente y los pasa al metodo predecirDieta() una vez tenemos la dieta del cliente. Obtiene una dieta del archivo json menus_por_dieta.json.</p> <p>Finalmente envia el resultado de la recomendación de la dieta, asi como un menu con Desayuno, Comida y Cena a la app Flutter para que esta lo muestre al usuario.</p>

Tabla 9: Tabla de metodos utilizados por la Aplicación

```
flutter: JSON enviado al servidor:
{
  "Activity_Level": 3.0,
  "Sodium_mg": 1500.0,
  "Sugar_g": 20.0,
  "Severity": 1.0,
  "Kidney Disease": 0,
  "Hypertension": 0,
  "Heart Disease": 0,
  "Diabetes": 1,
  "Obesity": 0
}
```

Figura 14: JSON enviado al API Flask

A continuación podemos observar una vista de la interfaz de usuario desarrollada en Flutter para la recolección de datos necesarios para la predicción dietética; ver Figura 15. Esta pantalla representa el primer punto de interacción entre el usuario y el sistema, permitiéndole ingresar

valores numéricos relacionados con su nivel de actividad, consumo de sodio, azúcar y severidad del estado clínico, así como seleccionar condiciones médicas relevantes como diabetes, obesidad o hipertensión a través de casillas de verificación. El diseño de la interfaz se ha enfocado en la simplicidad, claridad visual y facilidad de uso, empleando elementos gráficos intuitivos y etiquetas claras que guían al usuario durante el proceso de entrada de datos. Esta presentación estructurada reduce la posibilidad de errores de ingreso, mejora la accesibilidad y garantiza una experiencia coherente en el dispositivo móvil. Asimismo, este formulario está directamente conectado con el motor de predicción mediante llamadas a la API, asegurando que la información ingresada fluya eficientemente hacia el modelo de clasificación para obtener una recomendación personalizada.

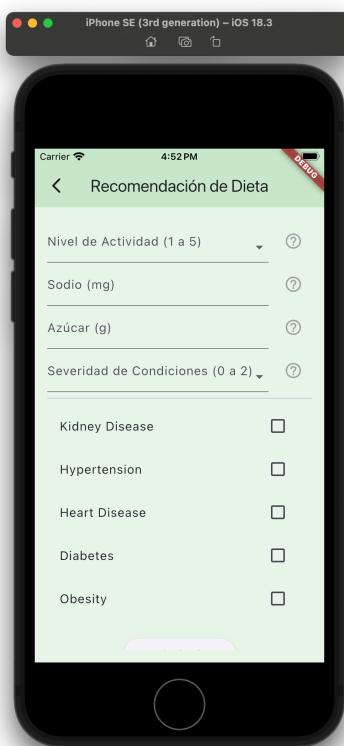


Figura 15: Imagen de la APP en Simulador

Una vez que la API recibe la información ingresada por el usuario, esta realiza la predicción utilizando el modelo MLP previamente entrenado y devuelve la respuesta a la aplicación

Flutter. Con base en esta predicción, la aplicación accede a un archivo JSON que contiene menús médicos predefinidos para cada tipo de dieta y selecciona aleatoriamente uno de los días disponibles. La Figura 16 muestra cómo se despliega esta información en la interfaz, presentando de forma clara y estructurada el menú correspondiente a un día específico dentro de la dieta recomendada, en este caso *Low-Glycemic*. Cada sección del menú incluye una descripción detallada del desayuno, comida y cena, acompañada de íconos representativos que facilitan su comprensión visual. Esta pantalla no solo refleja la recomendación generada por el modelo, sino que también proporciona al usuario una guía práctica y directa sobre qué alimentos consumir, fortaleciendo el valor aplicado del sistema. El diseño limpio y centrado en la experiencia del usuario garantiza que las recomendaciones sean fácilmente interpretables, fomentando la adherencia a los hábitos alimenticios sugeridos y cerrando el ciclo completo de interacción entre el modelo predictivo y su implementación funcional dentro de la app móvil.



Figura 16: Recomendación al Usuario

CONCLUSIONES

El desarrollo de la aplicación para recomendación de dietas con IA representa una propuesta innovadora en el ámbito de la nutrición clínica personalizada, integrando técnicas de inteligencia artificial con un enfoque centrado en el usuario. A través del desarrollo de una app móvil respaldada por modelos de machine learning, se logra atender una necesidad crítica por las opciones del mercado actual, como lo es la generación de planes alimenticios personalizados basados en las condiciones médicas de los usuarios.

A diferencia de aplicaciones móviles convencionales, cuyo enfoque suele centrarse en metas genéricas como pérdida de peso o conteo calórico, el proyecto propuesto se alinea con una visión médica más compleja, incorporando condiciones como diabetes, hipertensión, obesidad, enfermedad renal y cardiaca como variables centrales a la hora de generar la recomendación.

Desde el punto de vista profesional, esta propuesta refleja el gran potencial que tiene la inteligencia artificial para cambiar el área de la salud, especialmente en países como Ecuador, donde la oferta de soluciones digitales clínicas personalizadas es limitada. Desde el punto de vista de diseño, la implementación en Flutter permitió crear una aplicación multiplataforma ligera y funcional, capaz de adaptarse a diversos dispositivos móviles. La integración con un backend desarrollado en Flask aseguró la escalabilidad del sistema y la correcta comunicación entre el modelo predictivo y la interfaz de usuario. La visualización clara de los menús diarios, generados a partir de un archivo JSON, cierra el ciclo de personalización y entrega un valor práctico concreto a los usuarios. Otro uso que se le puede dar a la aplicación por parte de los profesionales de la salud es como apoyo a la hora de asignar dietas a sus pacientes. A nivel internacional, el proyecto se alinea con tendencias globales que buscan digitalizar la salud, consolidando un aporte que puede escalar y adaptarse a distintos contextos sanitarios.

A pesar de todo lo positivo, durante el desarrollo del proyecto se identificaron desafíos importantes, especialmente en la obtención de un dataset para realizar el entrenamiento de los modelos, debido a la falta de datasets de acceso público fiables que relacionen condiciones clínicas con recomendaciones alimenticias detalladas. Este problema fue abordado mediante la síntesis de literatura científica especializada, lo cual nos permitió construir un dataset sólido y médicaamente fundamentado. También se enfrentaron a dificultades en la integración del modelo con la interfaz de Flutter, especialmente la compatibilidad entre el modelo, los

encoders y el API Flask, lo cual requirió un trabajo de ajuste en la parte de entrega y recibo de datos.

Como proyección a futuro, se plantea incorporar un sistema de retroalimentación continua por parte del usuario que permita actualizar y ajustar los modelos según el contexto y la adherencia real. Asimismo, sería interesante evaluar la integración de dispositivos de monitoreo clínico como glucómetros, tensiómetros, entre otros, y extender el modelo para que realice recomendaciones más complejas, como combinaciones de recetas, alertas de cumplimiento o incluso sugerencias basadas en la disponibilidad de alimentos locales. Así mismo, sería valioso validar el sistema con la colaboración de profesionales de la salud, a fin de convertir la herramienta en un instrumento clínico con respaldo profesional. Además, se podría explorar el uso de Vibe Coding, un enfoque emergente que aprovecha modelos de inteligencia artificial generativa para crear aplicaciones a partir de instrucciones en lenguaje natural (*prompts*). Herramientas como Cursor permiten que desarrolladores diseñen interfaces o funcionalidades rápidamente mediante descripciones textuales, lo cual podría acelerar el prototipado, facilitar pruebas de concepto e incluso permitir la personalización dinámica de funcionalidades en futuras versiones de la app.

En conclusión, este trabajo ha logrado sentar las bases de una solución innovadora y viable para el problema de la personalización de planes alimenticios en contextos clínicos. Su enfoque técnico, su solidez metodológica y su orientación práctica lo convierten en un aporte relevante para el avance de la nutrición digital tanto en el contexto nacional como internacional.

REFERENCIAS

- [1] I. Callejas Gil, C. B. Moreno, R. M. Botaya, M. B. Otal, R. B. Andrés, y M. V. Colás, “Aplicación de Las Nuevas Tecnologías En Atención Primaria En El Control Del Paciente Con Sobrepeso u Obesidad y Factores de Riesgo Cardiovascular,” *Revista Clínica de Medicina de Familia*, 2023.
- [2] M. Franco, R. S. Cooper, U. Bilal, y V. Fuster, “Control of Coronary Heart Disease Risk Factors and Evidence-Based Therapies: Joint Efforts for Coronary Heart Disease Prevention in Spain,” *Revista Española de Cardiología*, 2011.
- [3] D. McKensy-Sambola, M. A. Rodríguez-García, F. García-Sánchez, y R. Valencia-García, “Ontology-Based Nutritional Recommender System,” *Applied Sciences*, vol. 12, no. 1, 2021.
- [4] A. M. Villar Jiménez, J. Matzumura-Kasano, y H. Gutiérrez-Crespo, “Teleorientación y Teleseguimiento En La Ingesta Alimentaria de Pacientes Con Hipertensión Arterial y Diabetes Mellitus,” *Horizonte Médico*, vol. 22, no. 1, 2022.
- [5] M. Barnett et al., “A Systematic Review and Meta-Analysis of Controlled Trials on Dietary Interventions via Digital Health Platforms in Chronic Conditions,” *Nutrients*, vol. 14, no. 5, 2022.
- [6] P. Chauveau et al., “Mediterranean Diet Interventions in Patients with Chronic Kidney Disease: A Review Article,” *Journal of Renal Nutrition*, vol. 28, no. 4, 2018.
- [7] J. W. Anderson et al., “Systematic Review and Meta-Analysis of High-Fiber, High-Carbohydrate, and Low-Glycemic Index Diets in Individuals with Diabetes,” *Nutrition Reviews*, vol. 62, no. 5, 2004.
- [8] L. J. Appel et al., “Randomized Crossover Feeding Study on Carbohydrate-, Protein-, and Unsaturated Fat–Rich Diets in Adults with Prehypertension or Stage 1 Hypertension,” *JAMA*, vol. 294, no. 19, 2005.

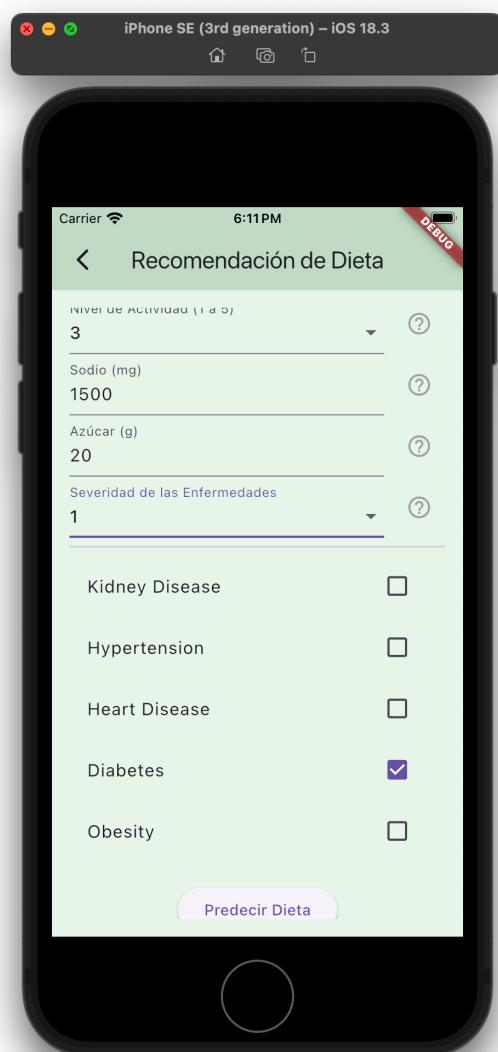
- [9] P. J. Elmer et al., “*Randomized Controlled Trial on the DASH Diet in Adults with Prehypertension or Stage 1 Hypertension*,” *Annals of Internal Medicine*, vol. 144, no. 7, 2006.
- [10] J. McMillan-Price et al., “*RCT Comparing High-Carb and High-Protein Diets (High vs. Low GI) in Overweight or Obese Young Adults*,” *Archives of Internal Medicine*, vol. 166, no. 14, 2006.
- [11] W. S. Yancy et al., “*Randomized Trial Comparing Low-Carbohydrate Ketogenic Diet and Low-Fat Diet in Hyperlipidemic Adults (BMI 30–60)*,” *Annals of Internal Medicine*, vol. 140, no. 10, 2004.
- [12] K. Esposito et al., “*Mediterranean-Style vs. Low-Fat Diet in Overweight Patients with Newly Diagnosed Type 2 Diabetes: A Randomized Trial*,” *Annals of Internal Medicine*, vol. 151, no. 5, 2009.
- [13] O. Ajala et al., “*Systematic Review and Meta-Analysis of Dietary Approaches for Type 2 Diabetes: Low-Carb, Vegan, Vegetarian, Mediterranean, Low-GI, and High-Protein Diets*,” *American Journal of Clinical Nutrition*, vol. 97, no. 3, 2013.
- [14] K. L. Campbell et al., “*Self-Managed Sodium Restriction in Adults with Hypertension and CKD: A Randomized Controlled Trial*,” *Kidney International*, vol. 91, no. 5, 2017.
- [15] S. Yoon et al., “*Prospective Cohort Study on Sodium Intake in 7,977 Individuals With and Without Hypertension*,” *Circulation*, vol. 135, no. 2, 2017.
- [16] D. P. Zaharieva and M. C. Riddell, “*Review of Low Glycemic Index Diets in Type 1 and Type 2 Diabetes Across 29 Trials*,” *Diabetes Spectrum*, vol. 36, no. 1, 2023.
- [17] A. E. A. Tito, B. O. H. Condori, Y. P. Vera, Universidad Nacional de San Agustín de Arequipa, “Análisis comparativo de técnicas de Machine Learning para la predicción de casos de deserción universitaria,” *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, núm. 51, pp. 84–98, 2023.
- [18] R. Singh, A. K. Dubey y R. Kapoor, “Deep Neural Network Regularization (DNNR) on Denoised Image,” *International Journal of Intelligent Information Technologies*, 2022.

- [19] J. A. Santos, A. Carrasco y J. F. Martínez, “Feature selection using typical testors applied to estimation of stellar parameters,” *Journal of Computación y Sistemas*, vol. 8, no. 4, pp. 294–300, 2004.
- [20] D. Nishad, A. Mishra y N. Goyal, “Symptom-Based Disease Prediction Using Machine Learning,” en *Proceedings of Confluence*, 2024.
- [21] A. Patil, N. Kardekar, G. Ghawade, R. Shirke y P. Khadse, “Integration of Flask and Python on the Face Recognition Based Attendance System,” en *Proceedings of the 2nd International Conference on Innovative and Creative Information Technology (ICIT)*, 2021.

ANEXO A: PAGINA DE INICIO DE LA APP



ANEXO B: EJEMPLO DE INGRESO DE DATOS DEL USUARIO



ANEXO C: RESULTADO DE LA RECOMENDACIÓN DE LA DIETA CON SU RESPECTIVO MENU



ANEXO D: LINK DEL REPOSITORIO DE GITHUB

<https://github.com/Fabro2604/Proyecto-Integrador>