

UCUENCA

Universidad de Cuenca

Facultad de Ingeniería

Carrera de Ingeniería en Ciencias de la Computación

Diseño e implementación de un agente conversacional para optimizar el consumo de medicamentos en adultos mayores en la ciudad de Cuenca

Trabajo de titulación previo a la obtención del título de Ingeniero en Ciencias de la Computación


Autores:

Carlos Sebastián Armijos Pulla

Juan Daniel Cambizaca Quinde

Director:

Jorge Mauricio Espinoza Mejía

ORCID:  0000-0003-0452-9721

Codirectora:

Victoria Abril Ulloa

Cuenca, Ecuador

2023-07-27

Resumen

Enfocada en la necesidad de prevenir las interacciones fármaco-fármaco y fármaco-alimento en la población de adultos mayores en Cuenca, este proyecto presenta el desarrollo de un agente conversacional implementado en una aplicación móvil. El estudio integra una base de datos sobre interacciones fármaco-fármaco y fármaco-alimento, junto con un algoritmo de recomendación basado en filtrado de contenido, diseñado específicamente para sugerir alternativas alimenticias en caso de interacciones potenciales entre un fármaco y un alimento. Los hallazgos, avalados por las evaluaciones realizadas tanto por estudiantes de nutrición como por expertos en el área, confirman la eficacia del chatbot en la detección de interacciones farmacológicas y la generación de recomendaciones adecuadas. Estos resultados resaltan la importancia de contar con sistemas de precisión que facilitan la interacción del usuario, por ejemplo, mediante agentes conversacionales y algoritmos de recomendación que proporcionen información personalizada para prevenir interacciones medicamentosas; además, la evaluación de expertos asegura la fiabilidad y eficacia de dichos sistemas.

Palabras clave: chatbot, filtrado basado en contenido, interacción fármaco-fármaco, interacción fármaco-alimento



El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Cuenca ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por la propiedad intelectual y los derechos de autor.

Repositorio Institucional: <https://dspace.ucuenca.edu.ec/>

Abstract

Centered on the need to prevent drug-drug and drug-food interactions among the elderly population in Cuenca, this project presents the development of a conversational agent implemented in a mobile application. The study incorporates a database on drug-drug and drug-food interactions, together with a content-based recommendation algorithm, specifically designed to suggest dietary alternatives in the event of potential interactions between a drug and a food item. The findings, endorsed by evaluations conducted by both nutrition students and field experts, confirm the efficacy of the chatbot in identifying pharmacological interactions and generating appropriate recommendations. These results underline the importance of precision systems that facilitate user interaction, for instance, through conversational agents and recommendation algorithms that provide personalized information to prevent drug interactions; moreover, the evaluation by experts assures the reliability and effectiveness of such systems.

Keywords: chatbot, content-based filtering, drug-drug interaction, drug-food interaction



The content of this work corresponds to the right of expression of the authors and does not compromise the institutional thinking of the University of Cuenca, nor does it release its responsibility before third parties. The authors assume responsibility for the intellectual property and copyrights.

Institutional Repository: <https://dspace.ucuenca.edu.ec/>

Índice de contenido

Resumen	2
Abstract	3
Capítulo 1 – INTRODUCCIÓN	13
1.1. Motivación y contexto	13
1.2. Planteamiento del problema	15
1.3. Solución propuesta.....	17
1.4. Objetivos	18
1.4.1. <i>General</i>	18
1.4.2. Específicos	19
Capítulo 2 – CONTEXTO TECNOLÓGICO	20
2.1. Historia y evolución de los agentes conversacionales ³	20
2.2. ¿Qué es un agente conversacional?	21
2.3. Funciones de los agentes conversacionales	22
2.4. Componentes de los agentes conversacionales	23
2.4.1. Reconocimiento de entrada	24
2.4.2. Compresión del lenguaje	24
2.4.3. Administrador de diálogo	24
2.4.4. Dominio específico de componentes.....	25
2.4.5. Generador de respuestas	25
2.4.6. Renderizado	25
2.5. Arquitectura de los agentes conversacionales	26
2.5.1. De acuerdo a la estrategia utilizada para gestionar los diálogos	26
2.5.2. De acuerdo al proceso de generación de respuestas	27
2.6. Propuesta de arquitectura del agente conversacional	29
2.6.1. Algoritmo de recomendación	30
2.6.2. Base de datos del dominio	30
Capítulo 3 – DISEÑO DEL AGENTE CONVERSACIONAL	32
3.1. Arquitectura del agente conversacional.....	32
3.2. Agente conversacional	34
3.2.1. Reconocimiento de entrada.....	35
3.2.2. Compresión del lenguaje	38
3.2.3. Generador de respuestas	44
3.2.4. Renderizado	45

3.3. Dominio específico de componentes	46
3.3.1. Extractor de información específica	46
3.3.2. Algoritmo de recomendaciones	64
3.4. API	64
3.4.1. Web Scraping	64
3.4.2. Agregar Interacciones farmacológicas	66
3.5. API Interfaz de usuario	67
3.6. Dispositivos externos	67
3.6.1. Aplicación Móvil	68
3.6.2. Web	68
3.6.3. Alexa	69
Capítulo 4 – DISEÑO DEL ALGORITMO DE RECOMENDACIÓN	71
4.1. Entrada	72
4.2. Consulta de datos	72
4.3. Preprocesamiento de datos	73
4.4. Entrenamiento del modelo	75
4.5. Generación de recomendaciones	77
4.6. Verificación de interacciones fármaco-alimento	80
4.7. Salida	83
Capítulo 5 – APLICACIÓN MÓVIL	85
5.1. Diseño de la interfaz de usuario	86
5.2. Limitaciones y desafíos del reconocimiento de voz en Android	95
Capítulo 6 – EVALUACIÓN Y RESULTADOS	97
6.1. Evaluación de usabilidad	97
6.1.1. Cuestionario de Usabilidad de Chatbot (CUQ)	97
6.2. Evaluación del juicio del experto	107
6.2.1. Escenario 1	108
6.2.2. Escenario 2	108
6.2.3. Escenario 3	109
6.2.4. Escenario 4	109
Capítulo 7 – DISCUSIÓN	111
7.1. Implicaciones y recomendaciones	112
7.2. Limitaciones del estudio	112
Capítulo 8 – CONCLUSIONES	114
8.1. Trabajos futuros	115

Referencias117

Anexos122

Índice de figuras

Figura 1 Componentes generales de los agentes conversacionales.	23
Figura 2 Adaptación de los componentes del agente conversacional.	29
Figura 3 Arquitectura del agente conversacional.	33
Figura 4 Métodos implementados para normalización del texto.	35
Figura 5 Frases de entrenamiento para el modo preventivo.	37
Figura 6 Frases de entrenamiento para el modo preventivo.	38
Figura 7 Definición de los flujos del agente conversacional para ejecutar recomendaciones.	39
Figura 8 Definición del escenario preventivo del agente conversacional.	40
Figura 9 Definición del escenario informativo del agente conversacional.	42
Figura 10 Flujo del agente conversacional.	44
Figura 11 Modelo de base de datos sobre alimentos, fármacos e interacciones.	48
Figura 12 Descripción de almacenamiento de datos en la tabla composición.	49
Figura 13 Descripción de almacenamiento de datos en la tabla grupo alimento.	50
Figura 14 Descripción de almacenamiento de datos en la tabla Fármaco.	50
Figura 15 Descripción de almacenamiento de datos en la tabla Fabricante.	51
Figura 16 Descripción de almacenamiento de datos en la tabla Interacción fármaco- alimento.	51
Figura 17 Descripción de almacenamiento de datos en la tabla interacción fármaco fármaco.	52
Figura 18 Categorías con respecto a la carga glucémica.	58
Figura 19 Niveles de identificación de interacción fármaco-alimento.	63
Figura 20 Script de implementación de web scrapping.	65
Figura 21 Ejemplo de la ejecución de web scrapping.	66
Figura 22 Endpoint para agregar medicamentos con sus interacciones.	66
Figura 23 Estructura JSON para el envío de datos.	66
Figura 24 API para la interfaz de usuario.	67
Figura 25 Implementación en la Web.	69
Figura 26 Script de conexión entre el agente y Alexa.	70
Figura 27 Fases de implementación y funcionamiento del algoritmo de recomendación.	71
Figura 28 Ejemplo de entrada del algoritmo de recomendación.	72
Figura 29 Código de implementación del método getGrupoAlimento().	72
Figura 30 Código de implementación del método getAlimentos().	73
Figura 31 Código de implementación del método getData().	74
Figura 32 Preprocesamiento de los datos (alimentos).	74

Figura 33 Estandarización de los datos.	75
Figura 34 Modelo de la red neuronal del algoritmo de recomendación.	76
Figura 35 Compilación del modelo del algoritmo de recomendación.	76
Figura 36 Entrenamiento del modelo del algoritmo de recomendación.	77
Figura 37 Cálculo de similitudes de alimentos mediante la similitud del coseno.	78
Figura 38 Lista de valores obtenidos mediante la similitud del coseno.	78
Figura 39 Selección y recomendación de alimentos similares al plátano.	79
Figura 40 Evaluación del algoritmo de recomendación.	79
Figura 41 Flujo de verificación de interacciones fármaco-alimento.	82
Figura 42 Salida del algoritmo de recomendación.	83
Figura 43 Código para generar la salida del algoritmo.	83
Figura 44 Flujo de intercambio de información.	86
Figura 45 Interfaz de usuario de bienvenida.	87
Figura 46 Autenticación del usuario.	88
Figura 47 Autenticación del usuario.	89
Figura 48 Autenticación del usuario.	90
Figura 49 Autenticación del usuario.	91
Figura 50 Interfaz de usuario, agente conversacional: modo informativo y preventivo.	92
Figura 51 Interfaz de usuario, agente conversacional: registro de alimentos.	93
Figura 52 Interfaz de usuario, notificaciones.	94
Figura 53 Interfaz de usuario, comidas.	95
Figura 54 Resultados de la pregunta 1 de la evaluación de usabilidad.	100
Figura 55 Resultados de la pregunta 3 de la evaluación de usabilidad.	100
Figura 56 Resultados de la pregunta 5 de la evaluación de usabilidad.	101
Figura 57 Resultados de la pregunta 7 de la evaluación de usabilidad.	101
Figura 58 Resultados de la pregunta 9 de la evaluación de usabilidad.	102
Figura 59 Resultados de la pregunta 11 de la evaluación de usabilidad.	102
Figura 60 Resultados de la pregunta 2 de la evaluación de usabilidad.	103
Figura 61 Resultados de la pregunta 4 de la evaluación de usabilidad.	104
Figura 62 Resultados de la pregunta 6 de la evaluación de usabilidad.	104
Figura 63 Resultados de la pregunta 8 de la evaluación de usabilidad.	105
Figura 64 Resultados de la pregunta 10 de la evaluación de usabilidad.	106
Figura 65 Resultados del cuestionario de usabilidad del chatbot.	107

Índice de tablas

Tabla 1 Definición de la entidad alimentos.....	43
Tabla 2 Definición de la entidad de medicamentos.	43
Tabla 3 Valores de índice glucémico y carga glucémica.	58
Tabla 4 Ejemplos de tipos de porciones de alimentos.....	59
Tabla 5 Cuestionario de usabilidad del sistema de recomendación.	98

Dedicatoria

Dedico este logro de manera especial a mi querida madre Cecilia, sin duda gracias a ella no sería lo que soy; tu amor incondicional ha logrado que llegue lejos. A mi padre Carlos y mis hermanas que han sabido estar en el proceso y me han apoyado siempre. A mis sobrinos Dante, Sthefano y Mia que son mi más grande inspiración y no podría faltar mi Missita. También a una persona que, si bien ya no está presente, todo este esfuerzo lo he hecho pensando en ella, mi papito José.

Carlos Sebastián Armijos Pulla

Dedicatoria

Dedico esta tesis a mis queridos padres, Luis y Claudia. Sin su amor, guía y apoyo incansables, no habría podido emprender este viaje académico. El valor de su sacrificio y su compromiso incondicional con mi educación han sido la brújula que me guio en esta labor. Ustedes han sido mi inspiración y mi motivación para seguir adelante, independientemente de las dificultades que he enfrentado. Este logro es tanto mío como suyo, ya que cada página de esta tesis lleva la marca de su amor y dedicación. Este trabajo es un homenaje a ustedes y a todo lo que han hecho por mí.

Juan Daniel Cambizaca Quinde

Agradecimientos

Agradecido siempre Dios que desde el inicio me ha dado la fortaleza y valentía para avanzar en este proceso. También con mi familia y amigos que siempre estuvieron presentes y de manera especial a Juan con el cual he tenido el agrado de compartir gran parte de este proceso académico y como no agradecer a Andrea que me ha mostrado su apoyo incondicional.

Carlos Sebastián Armijos Pulla

Mi sincero agradecimiento a Dios, mi guía y luz en este viaje de aprendizaje. Su sabiduría me ha guiado para poner en práctica el conocimiento adquirido durante estos años. Estoy agradecido con Verónica, cuya perspicacia y apoyo han sido esenciales en el desarrollo y perfeccionamiento de la aplicación móvil que es el corazón de este trabajo. No puedo dejar de mencionar a Carlos, mi compañero de tesis y amigo, cuya colaboración fue imprescindible para culminar con éxito este trabajo.

Juan Daniel Cambizaca Quinde

Nuestro más sincero agradecimiento al Ing. Mauricio Espinoza, Ph.D., nuestro director. Su sabiduría, experiencia y paciencia han sido fundamentales para guiarnos a lo largo de nuestro trabajo de titulación.

Así mismo, a la Dra. Victoria Abril, Ph.D., nuestra codirectora, por la predisposición que siempre tuvo con nosotros para que este trabajo salga adelante.

Capítulo 1 – INTRODUCCIÓN

El objetivo principal de este estudio es diseñar e implementar un agente conversacional con el propósito de optimizar el consumo¹ de medicamentos en adultos mayores con diabetes en la ciudad de Cuenca. El agente tiene como finalidad lograr equilibrio y estabilidad, además de brindar a los usuarios información precisa y personalizada en la ingesta de alimentos y medicamentos, previniendo posibles interacciones fármaco-fármaco y fármaco-alimento; así como posibles efectos adversos, lo cual, contribuirá a manejar de manera más efectiva la enfermedad y mantener un estilo de vida saludable.

1.1. Motivación y contexto

El envejecimiento es considerado un fenómeno global producido por la disminución de la tasa de mortalidad y el aumento de la esperanza de vida. De hecho, esta tendencia puede continuar debido a la mejora de las condiciones sociales de la población, los avances médicos, tecnológicos y la adopción de un estilo de vida más proactivo y saludable (Forttes, 2020). El envejecimiento de la población tiene implicaciones amplias y profundas en diversos ámbitos de la sociedad, tanto en términos económicos como culturales, ya que los adultos mayores suelen tener necesidades y prioridades distintas a las de los grupos etarios más jóvenes (Partida, 2022). Según la Encuesta de Condiciones de Trabajo y Salud (2022) la esperanza de vida en Ecuador ha aumentado de 74,5 años en 2010 a 77,6 años en 2020; también, el número de adultos mayores ha aumentado considerablemente, pasando de un total de 700.000 personas en 2011 a más de 1,5 millones en 2020.

El aumento de la población en adultos mayores incurre en numerosas implicaciones para la sociedad y el sistema de salud ecuatoriano, puesto que poseen un mayor riesgo de desarrollar enfermedades crónicas, discapacidades y otras condiciones de salud que pueden restringir su capacidad para ejercer actividades cotidianas y reducir su calidad de vida. Una de las enfermedades que más ocurrencia tiene en el territorio local, es la diabetes. La diabetes es una enfermedad crónica caracterizada por niveles elevados de azúcar en la sangre, originado por problemas en la producción de insulina, hormona encargada de regular el metabolismo de la glucosa en el organismo. La insulina es producida por el páncreas y ayuda a que la glucosa, principal fuente de energía para las células del organismo, ingrese a las células y se utilice de manera eficiente (Zavala & Fernández, 2018).

Para los adultos mayores, la diabetes representa un factor preocupante, ya que la edad es uno de los factores de riesgo para el desarrollo de la enfermedad.

¹Optimizar el consumo hace referencia a la prevención de las interacciones fármaco-alimento y fármaco-fármaco.

A medida que ocurre el proceso de envejecimiento, la probabilidad de que se experimenten cambios en el organismo es más alta, lo que puede afectar su capacidad para procesar la glucosa. Además, este grupo etario a menudo presenta otros problemas de salud, como presión arterial alta o enfermedades cardíacas que pueden incrementar el riesgo de la enfermedad (Partida, 2022).

Generalmente, la gran mayoría de casos de diabetes se asocian al tipo II, este tipo de diabetes es caracterizado por la resistencia del organismo a la insulina o también puede ocasionarse por la producción ineficiente de la misma, y, está relacionada con el estilo de vida, la herencia y el sedentarismo. Los síntomas asociados a esta patología en ocasiones resultan complicados de identificar, pues algunos de ellos pueden atribuirse a cambios relacionados al envejecimiento (Forttes, 2020).

En Ecuador, la diabetes es considerada importante dada su alta prevalencia e impacto en la salud pública. Según el Atlas de la Diabetes de la Federación Internacional de Diabetes (IDF), en 2021 se estima que hay 1.8 millones de adultos con diabetes en Ecuador, lo que representa una prevalencia del 5.9% en la población de 20 a 79 años. Además, Keays en su publicación del 2019, indica que la prevalencia podría ser mayor, entre el 7.8% y el 9.7%. La diabetes se ha posicionado como la segunda causa de mortalidad en mujeres y la tercera en hombres, lo que representa un problema de salud pública significativo; esta enfermedad, está asociada con un mayor riesgo de complicaciones cardiovasculares, renales, oculares y amputaciones, lo que aumenta aún más la carga en el sistema de salud y la calidad de vida de los pacientes. Por lo tanto, la prevención y el tratamiento efectivo de la diabetes son esenciales para reducir la morbilidad y mejorar la salud de la población en general (Ensanut-Ecu, 2012).

Dentro de su tratamiento, se utilizan varios tipos de medicamentos como la metformina, insulina, sulfonilureas, entre otros. Estos pueden resultar eficaces en el control del nivel de azúcar en la sangre y prevenir complicaciones a largo plazo. No obstante, como cualquier medicamento, se encuentran ligados a posibles interacciones con otros fármacos o alimentos que el paciente consume para otras patologías médicas; estas interacciones pueden ocurrir de diferentes maneras. Bressan, G. en el año 2020 afirma que la interacción fármaco-alimento es bidireccional, ya que los efectos esperados de un fármaco pueden ser modificados al administrarse de manera previa o conjunta con determinados alimentos.

Por su parte, la absorción de nutrientes puede limitarse por la dosificación previa o conjunta de un fármaco específico. En este punto, se menciona la interacción del medicamento sobre el alimento (IMA) y la interacción del alimento sobre el medicamento (IAM). Dentro de la IMA

se altera la utilización común de los nutrientes y el estado nutricional del paciente debido a la administración de fármacos, mientras que en la IAM los alimentos influyen sobre la respuesta farmacológica de la medicación, disminuyendo el efecto terapéutico. Por otro lado, la interacción fármaco-fármaco se refiere al caso en que dos o más medicamentos interactúan entre sí, alterando su efectividad o el riesgo de efectos secundarios. Esto puede ocurrir cuando los medicamentos afectan al organismo o cuando uno de los medicamentos afecta la manera en la que el cuerpo procesa el otro medicamento, por lo que es importante tener en cuenta las posibles interacciones entre medicamentos previo a su consumo (Vo et al., 2022).

Bressan, G. (2020) en su estudio indica que la población adulta mayor es considerada altamente vulnerable a dichas interacciones debido al consumo elevado de fármacos, tanto prescritos como automedicados, lo que puede dar lugar al fenómeno de polifarmacia. Además, afirma que la importancia de entender las distintas interacciones radica no sólo en prevenir aquellas que pueden ser perjudiciales o no deseadas, sino también en desarrollar estrategias para aprovechar las interacciones beneficiosas para el paciente; también, al evitar interacciones que disminuyen la absorción o biodisponibilidad de los medicamentos, se promueve el éxito del tratamiento farmacológico, al asegurar las concentraciones plasmáticas adecuadas del medicamento prescrito, sin disminuir su potencia o eficacia debido a cambios en la concentración o estructura causados por los alimentos.

Por tanto, es crucial que los pacientes proporcionen a su médico tratante, una lista completa de los medicamentos que estén consumiendo, incluyendo aquellos que han sido adquiridos sin prescripción médica, así como también los suplementos dietéticos. El profesional de la salud tendrá la capacidad de analizar las posibles interacciones que puedan existir entre los diversos medicamentos y, de ser necesario, hacer los ajustes pertinentes en cuanto a la dosificación o elección de los fármacos para prevenir o disminuir las interacciones y garantizar una atención médica segura y efectiva en la población de adultos mayores.

1.2. Planteamiento del problema

En Ecuador, al igual que en otras naciones, existe una distinción entre los fármacos de venta libre y aquellos que requieren receta médica. Los medicamentos de venta libre, también llamados medicamentos sin receta, son aquellos que se pueden obtener sin necesidad de una prescripción médica. Estos medicamentos suelen estar destinados al tratamiento de síntomas o afecciones leves y no representan un riesgo significativo de abuso o efectos adversos serios. Si bien estos fármacos suelen ser seguros cuando se emplean siguiendo las indicaciones del fabricante, su acceso sin receta médica también puede fomentar la automedicación. La automedicación, consiste en el uso de medicamentos sin prescripción

médica, esta práctica puede ser impulsada por diferentes razones, como la falta de acceso a servicios médicos, el deseo de ahorrar tiempo y dinero, o la creencia de que ciertos problemas de salud no requieren atención médica (Villalta et al., 2018). Si bien puede ser útil en algunos casos, como por ejemplo en el alivio temporal de síntomas menores.

La automedicación se ha convertido en una preocupación, puesto que puede ocasionar complicaciones de salud, como interacciones farmacológicas riesgosas, resistencia a los antibióticos y efectos secundarios negativos. Además, puede ocultar síntomas de patologías más complejas, lo que retrasa una atención médica adecuada y, en última instancia, puede deteriorar la salud del paciente (Navarrete, et al., 2020). A fin de asegurar el uso efectivo y seguro de los fármacos, es fundamental que el paciente esté informado sobre las indicaciones, contraindicaciones, dosis apropiadas y posibles efectos adversos de los medicamentos sin prescripción médica. Es por esto que, resulta importante que el personal farmacéutico y profesionales de la salud brinden asesoramiento a los pacientes con el fin de garantizar el uso adecuado de medicamentos sin receta y aquellos de prescripción médica en adultos mayores.

La administración de fármacos en adultos mayores conlleva una responsabilidad fundamental, en cuanto a la provisión de información, orientación sobre la correcta utilización y posibles efectos asociados a dichos medicamentos, la cual recae en el profesional médico encargado de prescribirlos. Azevedo y otros autores en un estudio de 2017 proponen que, durante una cita médica, los profesionales de la salud pueden usar señales verbales y no verbales; como el tono de voz y las expresiones faciales, para destacar la información más importante y ayudar al paciente a recordar la información médica.

Sin embargo, estas técnicas no siempre muestran el resultado esperado, tomando en cuenta que los pacientes son adultos mayores y podría considerarse que la información dada sea insuficiente o denote poca memoria por parte del paciente posterior a la cita médica. Por esta razón y gracias a los avances tecnológicos, se ha planteado la incorporación de agentes conversacionales, los cuales podrían contribuir a la disminución de esta problemática. Con ello, se obtendrían algunos beneficios para el adulto mayor, por ejemplo, i) cuántas veces al día debe tomar un medicamento, ii) qué cantidad debe tomar, iii) si es posible tomar el medicamento con o sin comida, iv) los posibles efectos de este medicamento, etc; lo cual puede optimizar la seguridad en la toma de medicamentos, todo esto gracias al disponer de un agente conversacional.

Ameur y Heudin (2006) definen al agente conversacional como “un programa de software diseñado para simular una conversación con un humano”. Los agentes conversacionales

pueden ser una herramienta valiosa para la integración de las demandas del usuario, los datos y servicios de salud, lo que facilita el acceso a la información sanitaria y promueve la toma de decisiones médicas. De hecho, Singh y otros autores en su publicación del 2022 desarrollaron un prototipo de agente conversacional para realizar consultas de los usuarios respecto a suplementos dietéticos. Por su parte, Goel (2021) propuso la construcción de dos tipos de agentes conversacionales que incorporan mecanismos de atención para generar respuestas empáticas para personas que padecen depresión. Asimismo, Arrabales (2020) desarrolló el agente conversacional 'Perla' capaz de realizar una entrevista a un paciente con depresión, el cual obtuvo gran aceptación por usuarios de internet. Además, Mugoye en una publicación realizada en 2019 discute sobre la necesidad de incluir chatbots o agentes conversacionales en los dispositivos inteligentes como manera de brindar apoyo a mujeres embarazadas.

Las publicaciones mencionadas representan utilidad cuando se realizan a través de soluciones informáticas que parten de modelos de representación de datos (Dingler et al., 2021). Por ejemplo, Preininger et al., (2020) describe la arquitectura de un sistema y su rendimiento en base al agente conversacional: Watson Assistant disponible en la base de datos Micromedex. Jia et al., (2015) plantea el desarrollo de un sistema de recomendación de atracciones turísticas basado en el usuario, utilizando la técnica de filtrado colaborativo, típicamente usado en este tipo de sistemas para proporcionar recomendaciones personalizadas a los usuarios; además, incluye una base de datos con las calificaciones de los usuarios y el histórico de visitas a un lugar determinado, los cuales son modelados y analizados por el algoritmo de recomendación colaborativa. Sin embargo, dado el avance de esta tecnología, aún no se contempla un agente conversacional basado en las interacciones fármaco-alimento y fármaco-fármaco.

1.3. Solución propuesta

Basado en las tendencias tecnológicas y la inclusión de nuevas bases de datos, algoritmos y asistentes virtuales, se pretende implementar un nuevo agente conversacional con un componente que brinde recomendaciones basadas en la interacción fármaco-alimento y fármaco-fármaco, previamente obtenidas en la página web DrugBank²; con lo cual, se mejorará la adherencia a los tratamientos, ya que el agente brindará recomendaciones y pautas para facilitar el seguimiento del mismo. Para este fin, se parte del modelado de una base de datos, alimentada con información que refleje las características de fármacos y alimentos.

²<https://go.drugbank.com/>

Debido al alcance planteado, se incluyen únicamente fármacos específicos para la diabetes; a partir de los medicamentos más utilizados o comunes, donde se extraerá una muestra de los mismos y los medicamentos que el usuario consuma para otras condiciones clínicas. En base a la muestra de medicamentos seleccionada, el sistema será capaz de realizar las recomendaciones enfocadas en dos escenarios: preventivo e informativo. Dentro del escenario preventivo, el usuario informa al agente la intención de tomar un medicamento diferente al que ha venido consumiendo, este analiza las interacciones fármaco-fármaco del medicamento y proporcionará una alerta si existen dichas interacciones.

En el contexto informativo, se enfoca en alertar al usuario acerca de las potenciales interacciones resultantes de la ingesta de alimentos o medicamentos, con vistas a una eventual interacción subsiguiente. Además, el agente se enfoca en advertir contraindicaciones, alternativas de consumo de alimentos y posibles interacciones fármaco-alimento y fármaco-fármaco, ya que estudios como el de Xiong et al., (2022) afirman que el uso simultáneo o sucesivo de varios medicamentos puede alterar la efectividad de un fármaco debido a la interacción con otros.

En la investigación de Liu et al. (2020) se afirma que las restricciones de tipo físico y fisiológico que experimentan los adultos mayores pueden obstaculizar la utilización de aplicaciones móviles, tales como la interacción con botones y el desplazamiento en la pantalla de los dispositivos inteligentes. Por esta razón, el uso de tecnologías como el reconocimiento de voz y del habla ha mejorado la facilidad en el uso de aplicaciones de salud diseñadas para este grupo etario.

El principal desafío en la creación del agente conversacional es la capacidad de acceder a la información almacenada en la base de datos a través de preguntas formuladas en lenguaje natural. Por lo que, la finalidad de este trabajo es obtener un agente conversacional integrado en una aplicación móvil, para garantizar la seguridad en el consumo de fármacos en adultos mayores. Además, se plantea realizar una evaluación de usabilidad y otra mediante la opinión de experto del campo de la salud, para medir el grado de precisión de las respuestas del agente conversacional frente a los criterios del experto tomando en cuenta escenarios previamente creados.

1.4. Objetivos

1.4.1. General

- Desarrollar un agente conversacional para optimizar el consumo de medicamentos en adultos mayores en la ciudad de Cuenca.

1.4.2. Específicos

- Diseñar una base de datos que permita almacenar información sobre fármacos, alimentos e información personal del usuario (adulto mayor).
- Generar un algoritmo de recomendación de alternativas de consumo de alimentos para evitar interacciones entre fármacos y alimentos.
- Implementar un prototipo de agente conversacional dentro de una aplicación móvil, para realizar las recomendaciones de las interacciones del fármaco con el paciente.

Capítulo 2 – CONTEXTO TECNOLÓGICO

En el presente capítulo, se abordarán conceptos claves para la comprensión de los agentes conversacionales, comenzando con un recorrido histórico que destaca su evolución y desarrollo desde sus inicios hasta su implementación contemporánea en una variedad de aplicaciones. Posteriormente, nos enfocamos en definir qué constituye un agente conversacional, estableciendo un marco para entender su habilidad única de interactuar y comunicarse de manera fluida con los humanos. Examinamos las funciones de estos agentes, destacando cómo manejan tareas complejas como el procesamiento de lenguaje natural, la comprensión del contexto y la generación de respuestas pertinentes. Concluimos con un análisis detallado de la arquitectura subyacente de estos agentes, poniendo de relieve la importancia y la interrelación de cada componente para proporcionar una experiencia de usuario efectiva y natural.

2.1. Historia y evolución de los agentes conversacionales³

A pesar del auge experimentado por los chatbots en los últimos años y las altas expectativas generadas en torno a ellos, su historia se remonta a 1964 con la creación de ELIZA por el científico informático Joseph Weizenbaum del MIT. En aquel entonces, ELIZA se consideraba la primera máquina capaz de mantener conversaciones utilizando los primeros avances en el procesamiento del lenguaje natural (NLP) (György & Zoltán, 2018).

ELIZA lograba engañar a muchas personas haciéndoles creer que estaban interactuando con un humano al adoptar el rol de una psicoterapeuta. Simplemente reemplazaba palabras en guiones y brindaba retroalimentación a los usuarios para mantener la conversación (Singh et al., 2018). Estos avances llevaron al desarrollo de un nuevo agente llamado PARRY en 1972, el cual presentaba ciertas aproximaciones basadas en la paranoia (Artificial Solutions, 2020).

A pesar de generar buenas expectativas, los primeros avances en el desarrollo de chatbots se estancaron debido al escaso progreso en el campo del NLP. El informe ALPAC de 1966 generó desmotivación e indiferencia en las investigaciones relacionadas con el NLP, y no se encontraron soluciones alternativas para obtener mejores resultados. Durante 20 años, hubo pocos avances significativos en este campo, con soluciones como Racter (1984), Jabberwacky (1988), Loebner Prize (1990) y Dr. Sbaits (1991) (Artificial Solutions, 2020).

Sin embargo, en 1995 se produjo una nueva aproximación más interesante en la creación de chatbots con ALICE (Artificial Linguistic Internet Computer Entity).

³Un agente conversacional es más conocido también como un agente virtual, asistente digital personales o incluso bots de chat (Microsoft, s.f).

ALICE se basó en el Lenguaje de Marcado de Inteligencia Artificial (AIML), que es muy similar a la estructura de las soluciones modernas de hoy en día (Singh et al., 2018). ALICE fue bien recibida y ganó en tres ocasiones el premio Loebner, que evalúa el software más inteligente presentado. Su implementación se basaba en la detección de patrones heurísticos para relacionar la entrada con reglas preestablecidas. Es importante destacar que ALICE tardó aproximadamente cinco años en ser liberada para su uso, y fue entre 2000 y 2004 cuando se pudieron observar sus resultados.

Los avances logrados durante este período demostraron la creciente necesidad de un desarrollo más enfocado en el uso de la inteligencia artificial, ya que el NLP por sí solo no permitía desarrollar una conversación fluida.

Por eso, gran parte del éxito de ALICE se debió a que, al inspirarse en ELIZA, llevó su experiencia más allá en una etapa temprana del manejo de técnicas de inteligencia artificial (Singh et al., 2018).

Continuando con el desarrollo e implementación de la inteligencia artificial y el aprendizaje automático en este campo, se llegó a Watson, un software desarrollado por IBM con el objetivo inicial de participar en el concurso de televisión "Jeopardy". Lo fundamental de Watson fue que IBM vio una gran oportunidad debido a los avances logrados en el área del aprendizaje automático para obtener buenos resultados. Desde entonces, Watson se ha convertido en un referente para la implementación de NLP y aprendizaje automático, permitiendo extraer información de grandes cantidades de datos y ampliando sus funcionalidades para convertirse en un asistente de inteligencia artificial para la automatización (Nuruzzaman & Hussain, 2018).

Watson refleja el fuerte interés en la creación de asistentes personales y el desarrollo de inteligencia artificial en la década de 2010, lo que permite interactuar de forma más natural con las máquinas. Es en este punto donde surgen soluciones más actuales como el asistente de Google, Cortana, Siri e incluso los chatbots creados por Facebook (Nuruzzaman & Hussain 2018). La rápida creación de chatbots ha generado expectativas muy altas, pero no han alcanzado los resultados esperados debido a la falta de claridad sobre cómo lograr una implementación adecuada de un sistema basado en modelos de aprendizaje automático con NLP. Esta sigue siendo la situación actual para la mayoría de los chatbots hasta la fecha.

2.2. ¿Qué es un agente conversacional?

Sin pretender dar una definición rigurosa, un agente conversacional es un sistema de software diseñado para interactuar y comunicarse con los usuarios de manera similar a como lo haría

un humano (Nuruzzaman & Hussain, 2018). Utiliza técnicas de inteligencia artificial y procesamiento del lenguaje natural para comprender y responder a las consultas y solicitudes de los usuarios.

Durante una conversación, la información se codifica y se divide en mensajes, que son unidades discretas. Este intercambio de mensajes ocurre en un medio o entorno donde los interlocutores están presentes. Los agentes perciben los mensajes dirigidos a ellos como estímulos provenientes del entorno y responden enviando mensajes al receptor correspondiente.

2.3. Funciones de los agentes conversacionales

López (2021) propone algunas funciones que deberían incorporar los agentes conversacionales, que, en conjunto, ofrecen una amplia gama de funcionalidades para mejorar la interacción y la experiencia de los usuarios.

- **Interacción y comunicación:** La función principal de un agente conversacional es establecer interacciones y comunicarse con usuarios o agentes externos. Esto implica recibir mensajes o entradas, comprender su contenido y generar respuestas adecuadas y pertinentes.
- **Resolución de consultas y preguntas:** Un agente conversacional tiene la capacidad de proporcionar información, responder preguntas y resolver consultas de los usuarios. Puede acceder a una base de conocimientos o realizar búsquedas en línea para brindar respuestas precisas y útiles.
- **Asistencia y soporte:** Un agente conversacional puede desempeñar el papel de asistente virtual o sistema de soporte, ofreciendo orientación, instrucciones, ayuda técnica u otro tipo de asistencia a los usuarios. Puede ayudar a los usuarios a navegar por un sitio web, realizar reservas, dar seguimiento a pedidos o solucionar problemas técnicos, entre otros.
- **Automatización de tareas:** Un agente conversacional puede llevar a cabo tareas automatizadas en nombre de los usuarios. Esto implica funciones como programar citas, enviar recordatorios, realizar transacciones, gestionar solicitudes u otras acciones que pueden realizarse a través de una interfaz conversacional.
- **Personalización y adaptabilidad:** Los agentes conversacionales pueden diseñarse para adaptarse y personalizarse según las preferencias y necesidades de los usuarios. Tienen la capacidad de aprender de interacciones previas, recopilar datos sobre los usuarios y utilizar técnicas de aprendizaje automático para ofrecer respuestas cada vez más precisas y personalizadas con el tiempo.

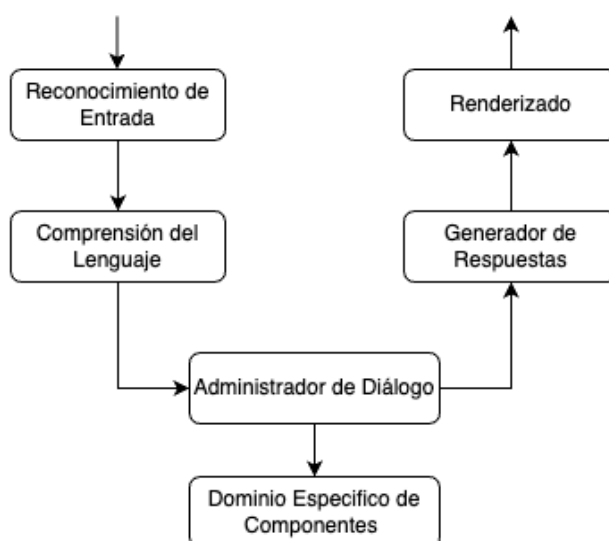
- Entretenimiento y experiencias interactivas: Algunos agentes conversacionales están diseñados para brindar entretenimiento, como juegos, historias interactivas o simulaciones. Pueden crear experiencias inmersivas y atractivas para los usuarios, ofreciendo diversión y entretenimiento.

En este trabajo, se ha adoptado una funcionalidad centrada en la resolución de consultas y preguntas relacionadas con las posibles interacciones entre fármacos-fármacos y fármacos-alimentos. Este enfoque se basa en aprovechar las capacidades de un agente conversacional para recibir consultas de usuarios y brindar respuestas precisas y útiles en el contexto de las interacciones entre fármacos y alimentos. Al utilizar una base de datos externa como fuente de información, se busca proporcionar a los usuarios una herramienta que les permita obtener información relevante y tomar decisiones informadas sobre el consumo de medicamentos y alimentos. En el capítulo 3, se explorará en mayor detalle los componentes tecnológicos que respalda esta funcionalidad y cómo se integra en el desarrollo de la solución propuesta.

2.4. Componentes de los agentes conversacionales

A pesar de que no existe consenso sobre los componentes que forman parte de un agente conversacional, la mayoría de propuestas coinciden en los siguientes: el reconocimiento de entrada, la comprensión del lenguaje natural, la gestión del diálogo, la generación de respuestas y el renderizado (Arora et al., 2013). La Figura 1 muestra el flujo de estos componentes.

Figura 1 Componentes generales de los agentes conversacionales.



Nota: Componentes del sistema de diálogo, adaptado de (Arora et al., 2013).

2.4.1. Reconocimiento de entrada

Arora y otros autores (2013) en su artículo titulado 'Dialogue system: A brief review', proponen que el decodificador de entrada en un agente conversacional es responsable de reconocer y convertir la entrada en texto simple. Este componente es especialmente relevante en aquellos agentes que no se basan en texto. El proceso de decodificación implica la conversión de sonidos hablados, es decir, las expresiones del usuario, en texto. Para llevar a cabo esta tarea, se requiere un conocimiento de fonética y fonología. Este componente se encuentra presente únicamente en sistemas de diálogo que no se basan en texto.

2.4.2. Compresión del lenguaje

El componente denominado *Compresión del Lenguaje* en los sistemas de diálogo desempeña un papel crucial en la comprensión del mensaje deseado por el usuario. Su función principal consiste en convertir una secuencia de palabras en una representación semántica que posteriormente puede ser utilizada por el gestor de diálogo. Este componente requiere el uso de diversos aspectos lingüísticos, como la morfología, la sintaxis y la semántica (Arora et al., 2013).

En este proceso, juega un papel esencial la comprensión de lenguaje natural (NLU, por sus siglas en inglés). NLU es una disciplina dentro del procesamiento del lenguaje natural (NLP) que se enfoca en entender la intención detrás de una oración y utiliza el aprendizaje activo para mejorar su comprensión de manera continua (López, 2021). Este componente es el responsable de dotar a las máquinas, como los chatbots, con la capacidad de comprender y procesar mensajes.

Una vez que el sistema ha logrado comprender la intención del usuario gracias al NLU, el NLP utiliza esta información para generar la respuesta apropiada. En este sentido, el NLU se enfoca en aplicar técnicas de aprendizaje automático para otorgar independencia a las máquinas en la comprensión del lenguaje humano. Esto se contrapone con el objetivo del NLP, que es la formalización del lenguaje a través de la conversión de texto en datos estructurados.

2.4.3. Administrador de diálogo

El gestor de diálogo tiene la responsabilidad de supervisar y controlar todos los aspectos de la conversación en un agente conversacional. Este componente utiliza una representación semántica del texto del usuario para comprender cómo encaja dentro del contexto global y genera una representación semántica de la respuesta del sistema. También, desempeña una amplia gama de tareas, que incluyen el registro histórico del diálogo, la adopción de

estrategias específicas para la interacción, el manejo de texto que no cumple con la estructura esperada o que no puede ser reconocido, la recuperación de información almacenada en archivos o bases de datos, la determinación de la mejor respuesta para el usuario, la gestión de la iniciativa y las respuestas del sistema, el abordaje de aspectos de pragmática en la comunicación, el análisis del discurso, y el establecimiento de una base común de conocimiento y comprensión (Arora et al., 2013).

Para llevar a cabo estas tareas, el gestor de diálogo se compone de varios componentes, tales como el modelo de diálogo, el modelo de usuario, la base de conocimiento, el gestor de discurso, el resolver de referencias y el módulo de establecimiento de una base común de conocimiento y comprensión.

2.4.4. Dominio específico de componentes

El gestor de diálogo generalmente necesita interactuar con algún software externo, como una base de datos. Por lo tanto, las consultas deben convertirse de la representación interna utilizada por el gestor de diálogo al formato utilizado por el sistema específico del dominio externo (por ejemplo, SQL). Esta interfaz es manejada por los componentes específicos del dominio (Arora et al., 2013).

2.4.5. Generador de respuestas

Este componente implica la generación del mensaje que será emitido para el usuario. Se encarga de tomar decisiones en cuanto a qué información debe ser incluida, cómo debe ser estructurada, la elección de las palabras y la estructura sintáctica del mensaje. En los sistemas actuales, se emplean métodos sencillos como la inserción de datos recuperados en espacios predefinidos dentro de una plantilla (Arora et al., 2013).

2.4.6. Renderizado

Este componente desempeña la tarea de transformar el mensaje generado por el componente de generación de respuestas en una forma hablada. Para lograr la generación de voz, se emplean dos enfoques distintos. El primero se basa en el uso de grabaciones preexistentes que contienen espacios en los cuales se insertan muestras recuperadas o grabaciones previas. Por ejemplo, podría utilizarse la frase 'Bienvenido, ¿en qué puedo ayudarte?'. El segundo enfoque se apoya en la síntesis de texto a voz, donde se genera habla a partir de texto mediante la conversión del texto a fonemas y, posteriormente, de los fonemas a habla. Este proceso se conoce como síntesis de habla concatenativa, conversión de texto a fonemas y fonemas a habla, o simplemente síntesis de texto a voz (Arora et al., 2013).

En este trabajo, se ha desarrollado un agente conversacional que utiliza varios componentes clave, entre los que incluyen el decodificador de entrada, que está enfocado a un modelo híbrido para brindar respuestas precisas y personalizadas a los usuarios, la comprensión del lenguaje, el gestor de diálogo, componentes específicos del dominio y un generador de respuestas. Su función conjunta es facilitar la extracción de información, procesar la entrada del usuario, gestionar el diálogo y generar respuestas adecuadas.

2.5. Arquitectura de los agentes conversacionales

Los agentes conversacionales pueden clasificarse tomando en cuenta diferentes aspectos:

2.5.1. De acuerdo a la estrategia utilizada para gestionar los diálogos

En base a la forma en cómo se gestionan los sistemas de diálogo pueden ser clasificados en tres categorías esenciales: sistemas fundamentados en estados finitos o gráficos, sistemas basados en marcos y sistemas centrados en agentes.

- Estados finitos: Los sistemas fundamentados en estados finitos se sustentan en la noción de proyectar el diálogo como un grupo limitado de estados y las transiciones entre estos. Estos sistemas se esquematizan por medio de gráficos que simbolizan las vías de interacción posibles en el diálogo (Arora et al., 2013).
- Basados en marcos: Los sistemas basados en marcos emplean una estrategia de llenado de plantillas derivada de las respuestas del usuario. En este tipo de sistemas, se plantean interrogantes al usuario que ayudan a llenar los espacios vacíos en una plantilla preestablecida para ejecutar tareas específicas. A diferencia de otros sistemas, el flujo del diálogo no está predefinido, sino que depende del contenido proporcionado por el usuario y de la información que el sistema necesita obtener (Arora et al., 2013).
- Basados en agentes: Los sistemas basados en agentes proporcionan un mecanismo de comunicación integral que permite la interacción entre el sistema, el usuario y la aplicación, con el propósito de abordar y solucionar tareas o problemas específicos. Esta interacción se concibe como un encuentro entre dos agentes autónomos, cada uno de ellos capaz de reflexionar sobre sus propias acciones y convicciones. El modelo de diálogo se enriquece y adapta en función del contexto anterior. Por lo tanto, a medida que se desarrolla el diálogo, éste se despliega en una secuencia dinámica de etapas interrelacionadas que se edifican progresivamente (Arora et al., 2013).

En la arquitectura de diálogo del presente trabajo, se implementan los tres enfoques para proporcionar una interacción óptima con el usuario. En primer lugar, se utilizarán los sistemas basados en estados finitos para estructurar el diálogo, modelando la conversación en términos de estados específicos y transiciones en respuesta a la entrada del usuario. En segundo lugar, se implementan los sistemas basados en marcos para manejar la información específica del usuario de forma eficiente. Estos marcos contienen datos específicos, como una lista de medicamentos, listas de alimentos que se llenarán durante la interacción con el usuario.

Finalmente, la arquitectura incorpora características de los sistemas basados en agentes para permitir una interacción autónoma y adaptable. Este enfoque permitirá al sistema buscar de manera inteligente interacciones farmacológicas en una base de datos basándose en los medicamentos ingresados por el usuario, y adaptarse a la entrada adicional del usuario para proporcionar información relevante y precisa. Esta combinación de enfoques proporcionará una estructura sólida para el diálogo, un manejo eficiente de la información y una interacción centrada en el usuario.

2.5.2. De acuerdo al proceso de generación de respuestas

El proceso de generación de respuestas puede ser guiado por una variedad de enfoques o modelos. Se destacan cinco tipos principales de modelos: modelos basados en recuerdos, modelos basados en reglas, modelos generativos, modelos basados en aprendizaje automático y modelos híbridos.

- **Basados en reglas:** Este modelo emplea reglas predefinidas para generar respuestas. Esta estrategia se basa en la creación de un conjunto de reglas que se aplican a las entradas del usuario para determinar la respuesta adecuada. Aunque es uno de los modelos más antiguos y simples, se utiliza con frecuencia en chatbots con un conjunto limitado de respuestas posibles. Sin embargo, este enfoque presenta limitaciones en su capacidad para lidiar con la ambigüedad y la complejidad del lenguaje natural (Allouch et al., 2021).

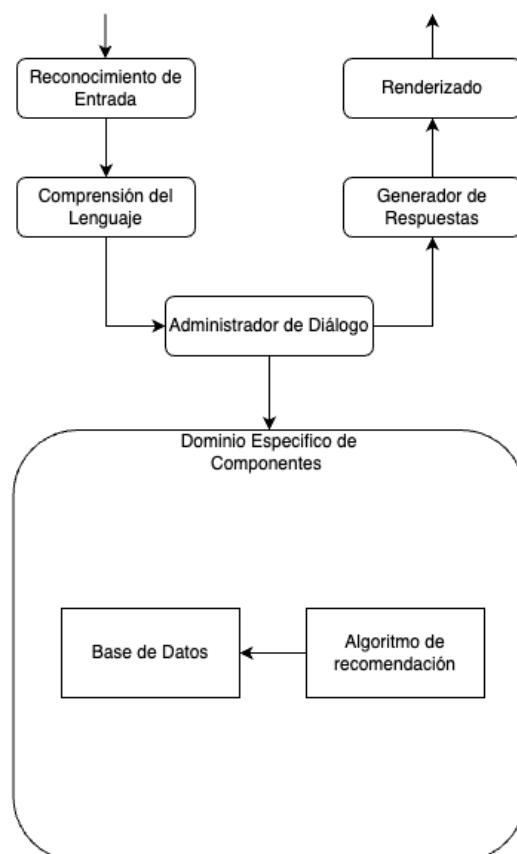
- **Basados en recuerdos:** El modelo basado en recuerdos utiliza técnicas de recuperación de información para generar respuestas. Este enfoque se fundamenta en la búsqueda y recuperación de respuestas previas almacenadas en una base de datos. El modelo utiliza una base de datos de preguntas y respuestas anteriores para encontrar la respuesta más apropiada a la pregunta actual del usuario. Esta estrategia es comúnmente empleada en chatbots con un conjunto limitado de posibles respuestas y que se enfocan en la recuperación de información específica (Allouch et al., 2021).
- **Modelos generativos:** El modelo generativo utiliza técnicas de aprendizaje profundo para generar respuestas. A diferencia de simplemente seleccionar una respuesta preexistente de una base de datos, este enfoque se enfoca en crear un modelo capaz de generar respuestas nuevas y originales. El modelo generativo utiliza redes neuronales y aprendizaje contextual para generar respuestas basadas en el contexto de la conversación. Esta estrategia es más flexible en comparación con los modelos basados en reglas o en recuperación, ya que no se limita a un conjunto predefinido de reglas o respuestas, lo que permite una mayor diversidad en las respuestas generadas (Allouch et al., 2021).
- **Modelos basados en aprendizaje automático:** El modelo basado en aprendizaje automático utiliza técnicas de aprendizaje automático para mejorar la capacidad de respuesta de los chatbots. Este enfoque se basa en la creación de un modelo capaz de aprender de los datos de entrenamiento, mejorando así su capacidad para generar respuestas precisas y personalizadas. A través de técnicas de redes neuronales, el modelo adquiere la capacidad de generar respuestas contextualizadas y se adapta a medida que interactúa con los usuarios. A diferencia de los modelos basados en reglas o recuperación, este enfoque ofrece mayor flexibilidad al generar respuestas que no están limitadas por un conjunto predefinido de reglas o respuestas (Nuruzzaman & Hussain, 2017).
- **Modelos híbridos:** El modelo híbrido combina varios enfoques, como modelos basados en reglas, modelos basados en recuperación y modelos generativos. Este enfoque aprovecha las fortalezas de cada modelo y supera sus limitaciones. Por ejemplo, puede utilizar un modelo basado en reglas para preguntas simples y comunes, uno basado en recuperación para preguntas más específicas, y uno generativo para preguntas complejas y abiertas. Los modelos híbridos son cada vez más populares en la industria de los chatbots, ya que brindan respuestas precisas y personalizadas a los usuarios de manera efectiva (Allouch et al., 2021).

Para este trabajo, se ha utilizado principalmente modelos basados en reglas y modelos basados en aprendizaje automático para lograr una interacción efectiva con el usuario. Los modelos basados en reglas permiten establecer el flujo de la conversación con acciones y respuestas predefinidas según las entradas del usuario. Por otro lado, los modelos basados en aprendizaje automático se entrenan con datos históricos para comprender las intenciones del usuario y ofrecer respuestas más precisas y personalizadas. Además, esta combinación de enfoques nos permite aprovechar las ventajas de ambos paradigmas, lo que resulta en un sistema híbrido que se adapta y mejora a medida que interactúa con los usuarios.

2.6. Propuesta de arquitectura del agente conversacional

A partir de los componentes que forman parte de un agente conversacional clásico, se ha visto la necesidad de incluir dos componentes extras: bases de datos del dominio y el algoritmo de recomendación. La Figura 2 ilustra la arquitectura del agente conversacional propuesto en este trabajo, incluyendo los componentes agregados para cumplir los objetivos del proyecto

Figura 2 Adaptación de los componentes del agente conversacional.



Nota: Componentes del sistema de diálogo, adaptado de (Arora et al., 2013).

2.6.1. Algoritmo de recomendación

El algoritmo de recomendación tiene como objetivo principal sugerir opciones o recomendaciones relevantes para los usuarios durante la interacción. En nuestra propuesta el algoritmo utiliza técnicas de basado en el contenido, en lugar de técnicas de filtrado colaborativo o híbridas para analizar los datos disponibles y generar recomendaciones personalizadas.

El algoritmo de recomendación se integra en la arquitectura del agente conversacional y se activa en momentos específicos de la conversación en función del contexto y las necesidades del usuario. Algunos de los roles y funciones que desempeña el algoritmo en la arquitectura de un agente conversacional implementado son:

- **Recopilación y análisis de datos:** El algoritmo de recomendación recopila y analiza datos relevantes sobre los alimentos que el usuario está por consumir, enfocándose en sus atributos nutricionales e información sobre posibles interacciones fármaco-alimento. Los datos son utilizados para identificar patrones y tendencias, lo que permite al algoritmo generar recomendaciones precisas y útiles.
- **Personalización basada en medicación:** El algoritmo de recomendación considera cuidadosamente la medicación que el usuario toma, analizando en detalle la información sobre posibles interacciones con los alimentos. Este análisis permite ofrecer recomendaciones personalizadas que se ajustan a las necesidades y restricciones específicas del usuario, asegurando la seguridad y la compatibilidad de los alimentos sugeridos.

2.6.2. Base de datos del dominio

Por otra parte, la base de datos desempeña un papel fundamental al almacenar y gestionar la información relevante para el agente. Actúa como un repositorio centralizado de datos que el agente puede utilizar durante las interacciones con los usuarios. Algunas de las funciones que realiza este componente en la arquitectura del agente conversacional son las siguientes:

- **Almacenamiento de información:** La base de datos almacena datos y conocimientos relevantes para el agente conversacional. Esto puede incluir información sobre alimentos, fármacos, interacciones, entre otros. La información almacenada en la base de datos actúa como una fuente de referencia para el agente.

- **Recuperación de información:** Durante una conversación con el usuario, el agente puede acceder a la base de datos para recuperar la información relevante sobre posibles interacciones fármaco-fármaco o fármaco nutriente. Esto le permite proporcionar respuestas precisas y actualizadas basadas en datos almacenados. Por ejemplo, si un usuario pregunta sobre si es posible comer un determinado alimento, el agente puede consultar la base de datos para recuperar esa información y proporcionar una respuesta precisa sobre si existe o no una posible interacción.
- **Actualización de datos:** La base de datos también permite la actualización de sus datos, cuya función la realiza únicamente su administrador. Por ejemplo, si se requiere añadir un alimento o fármaco determinado, la base de datos puede actualizarse para reflejar esta acción y mantener un registro actualizado. Esto asegura que el agente tenga acceso a la información más reciente durante las interacciones posteriores.

En el Capítulo 3, se ofrecerán los detalles del diseño de los componentes principales del agente conversacional implementado, mientras que en el Capítulo 4, se describe el algoritmo de recomendación utilizado para proporcionar alternativas de alimentos a partir de posibles interacciones.

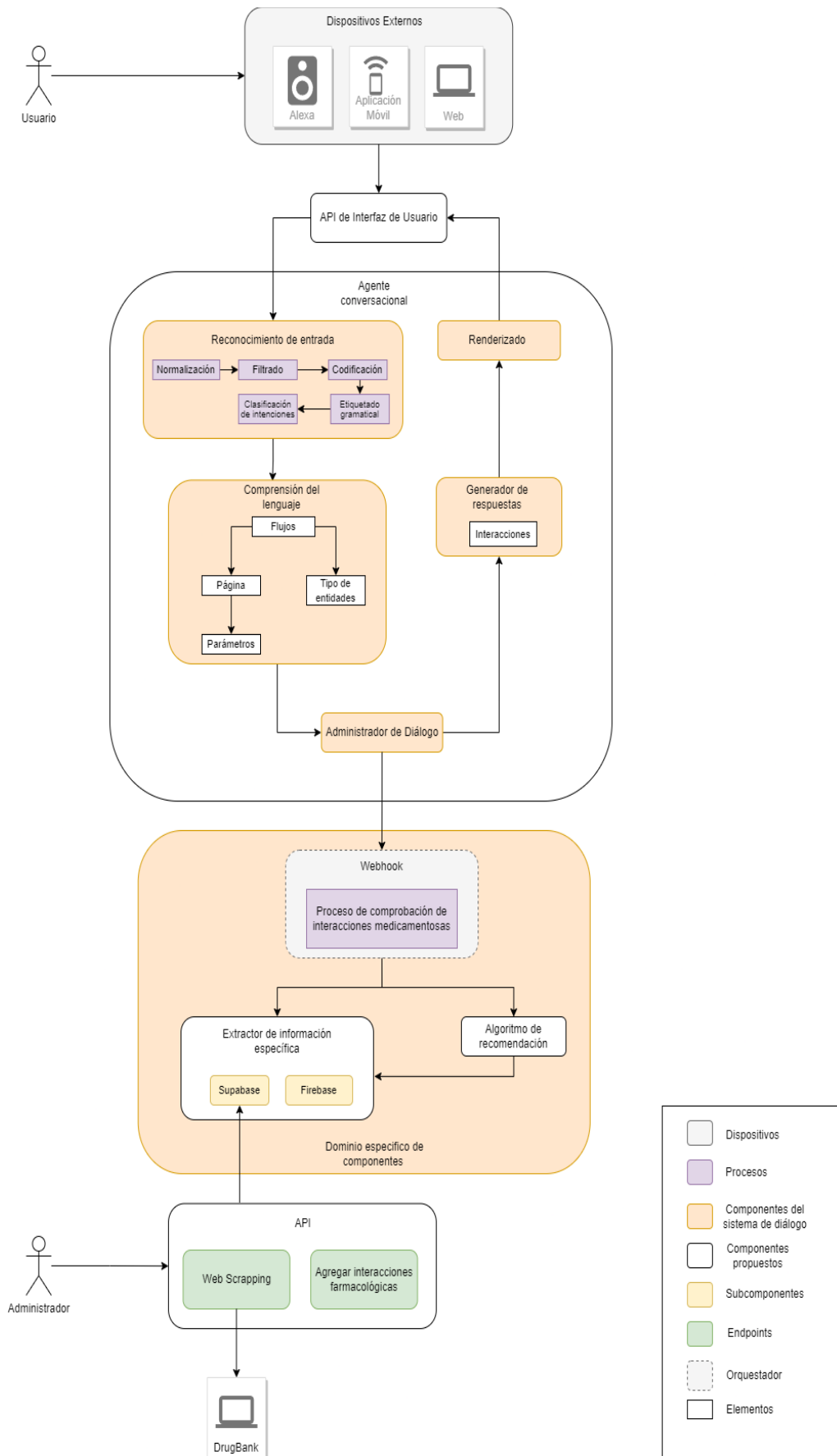
Capítulo 3 – DISEÑO DEL AGENTE CONVERSACIONAL

En este capítulo, se describe en detalle la arquitectura del agente conversacional, sus componentes y funcionamiento. También se detalla la implementación de la base de datos para las interacciones fármaco-fármaco y fármaco-alimento.

3.1. Arquitectura del agente conversacional

La arquitectura del agente conversacional propuesto se basa en los componentes descritos a continuación, en orden de importancia (Figura 3):

Figura 3 Arquitectura del agente conversacional.



3.2. Agente conversacional

Este componente, es el núcleo de la arquitectura, se encarga de manejar la interacción entre el usuario y el agente conversacional. El agente conversacional procesa las entradas del usuario, consulta la base de datos y el algoritmo de recomendación a través del webhook para determinar la respuesta más adecuada, y comunica dicha respuesta al usuario a través de la API de Interfaz de Usuario. Para implementar este componente, se ha seleccionado Dialogflow CX (Customer Experience) de Google, una plataforma robusta y versátil para la construcción de chatbots y sistemas de diálogo.

Dialogflow CX fundamenta su enfoque en sistemas 'basados en estados finitos', donde los 'estados' representan las diversas etapas de una conversación y las 'transiciones' determinan los posibles caminos conversacionales a seguir en función de las entradas del usuario.

Es relevante destacar que, a pesar de tener su base en una máquina de estados, Dialogflow CX integra componentes de otros paradigmas arquitectónicos para potenciar su funcionalidad. Un ejemplo claro de esto es su capacidad para utilizar entidades y parámetros, que son similares a los 'marcos' en un sistema basado en marcos (sección 2.5.1). Estos elementos permiten extraer y procesar información de las entradas del usuario de manera eficiente.

Además, se puede observar la influencia de los 'sistemas basados en agentes' en Dialogflow CX (sección 2.5.1), especialmente al implementar un agente de recomendación. Dicho agente tiene la habilidad de interactuar de forma más autónoma y adaptativa con el usuario, a pesar de que la lógica de la conversación se base en estados y transiciones.

Por otra parte, Dialogflow CX emplea un modelo híbrido para el reconocimiento de entradas. Este modelo híbrido es una amalgama de reglas predefinidas, recuperación basada en patrones y técnicas de aprendizaje automático. Las reglas se utilizan para manejar interacciones sencillas y directas, la recuperación permite buscar respuestas predefinidas en una base de datos mediante algoritmos de búsqueda y coincidencia de patrones. Por otro lado, el aprendizaje automático, que abarca procesamiento del lenguaje natural y redes neuronales, mejora la comprensión y generación de respuestas al adaptarse a variados contextos y cambios en el lenguaje.

Aparte de su modelo híbrido, Dialogflow CX destaca por su compatibilidad con múltiples plataformas, la integración con servicios de terceros, la implementación de procesamiento del lenguaje natural, y su escalabilidad que facilita el manejo de grandes volúmenes de datos. Estas ventajas, combinadas con las características y funcionalidades inherentes a los

agentes conversacionales, hacen que Dialogflow CX se constituya como la elección arquitectónica idónea para el desarrollo del agente.

3.2.1. Reconocimiento de entrada

El subcomponente Reconocimiento de Entrada es el responsable de realizar una serie de procesos fundamentales de PLN, que incluyen la normalización, el filtrado, el etiquetado gramatical, la codificación y la clasificación de intenciones.

- Normalización

La normalización es un paso esencial para asegurar que las entradas de texto de los usuarios sean interpretadas de manera coherente, independientemente de las variaciones en la forma y el estilo. Para solventar estas consideraciones de normalización, se ha desarrollado un script que realiza las transformaciones necesarias a los datos entrantes. Este script se encarga de cambiar la primera letra a mayúsculas, así como de convertir los términos a su forma singular. El código presentado en la Figura 4 representa una implementación inicial que, si bien no cubre todos los posibles escenarios, proporciona una base sólida sobre la cual se puede extender para manejar situaciones más complejas.

Figura 4 Métodos implementados para normalización del texto.

```

cambiarMayuscula(nombre) {
  const data = nombre.charAt(0).toUpperCase() + nombre.slice(1).toLowerCase();
  const data_singular = this.validarSingular(data);
  return data_singular;
}

validarSingular(nombre) {
  const ultimoCaracter = nombre.charAt(nombre.length - 1);
  const penultimoCaracter = nombre.charAt(nombre.length - 2);
  if (ultimoCaracter === 's') {
    if (penultimoCaracter === 'e') {
      return nombre.slice(0, -2);
    }
    return nombre.slice(0, -1);
  }
  return nombre;
}

validarSingularPlural(nombre) {
  const ultimoCaracter = nombre.charAt(nombre.length - 1);
  if (["a", "e", "i", "o", "u", "á", "é", "í", "ó", "ú", "n", "r"].includes(ultimoCaracter)) {
    return `${nombre}s`;
  } else {
    return `${nombre}es`;
  }
}

```

- Filtrado

El filtrado permite a Dialogflow CX centrarse en las palabras y frases que aportan un significado valioso a la conversación, eliminando las palabras vacías que no aportan a la semántica. Este proceso implica la asignación de etiquetas gramaticales a cada palabra en una oración, identificando si una palabra es un sustantivo, un verbo, un adjetivo, etc. Esto es fundamental para la capacidad de Dialogflow CX de interpretar las solicitudes de los usuarios y responder a ellas de manera precisa y efectiva.

- Codificación

La codificación convierte las palabras y las frases en vectores numéricos que las máquinas pueden entender. Dialogflow CX usa modelos que generan representaciones vectoriales de palabras que tienen en cuenta el contexto completo de una oración, incluyendo el orden de las palabras y las relaciones gramaticales. Estas representaciones son altamente efectivas para capturar los matices del lenguaje y mejorar la precisión de las tareas de comprensión del lenguaje natural.

- Etiquetado Gramatical

El etiquetado gramatical es fundamental para entender la estructura de las oraciones y la función de las palabras en ellas. Este proceso implica la asignación de etiquetas gramaticales a cada palabra en una oración, identificando si una palabra es un sustantivo, un verbo, un adjetivo, etc. De esta manera se puede usar en el agente frases que van de acuerdo a los escenarios: preventivo e informativo que se detallarán más adelante.

En el contexto de esta investigación, se busca promover un comportamiento preventivo e informativo en el agente de chatbot. Con este objetivo, se ha definido y aplicado un conjunto de frases representativas para entrenar el modelo con estos dos tipos de comportamientos.

Las frases mostradas en la Figura 5, se utilizaron para entrenar el modelo en modo preventivo para su activación. Estas sentencias abarcan una gama de formulaciones de solicitudes de información relacionadas con medicamentos y alimentos que el usuario esté pensando en consumir.

Por otro lado, las frases incluidas en la Figura 6, se utilizaron para entrenar el modelo en el modo informativo para su activación. Este grupo incluye consultas que solicitan asesoramiento o información sobre el estado del usuario, permitiendo al agente conversacional encontrar si algún medicamento o alimento consumido recientemente, es el causante del malestar del usuario. Ambos modos son explicados más adelante.

Es crucial notar que las frases utilizadas no son demasiado complejas ni extremadamente específicas, ya que se desea evitar el sobreajuste, que es cuando un modelo de aprendizaje automático se ajusta demasiado a los datos de entrenamiento y falla en generalizar a datos no vistos. Cabe resaltar que, dentro de cada modo, existen de igual forma frases que han sido entrenados en el agente conversacional para el flujo completo de la conversación.

Gracias al uso de técnicas de procesamiento del lenguaje natural, como el etiquetado gramatical y el reconocimiento de la estructura de la oración, el agente es capaz de comprender y procesar lo que el usuario comunica. Esta habilidad contribuye al desarrollo de un agente más efectivo y preciso, mejorando su rendimiento en términos de interacción con los usuarios.











Se ha probado estas frases de entrenamiento en el agente de manera iterativa, realizando ajustes cuando fue necesario, para optimizar su rendimiento. Esta estrategia de entrenamiento y la aplicación del NLP fortalecen la capacidad de respuesta del agente conversacional, permitiéndole manejar una gama más amplia de consultas de usuarios y evitar el sobreajuste.

Figura 5 Frases de entrenamiento para el modo preventivo.

Q Search Search training phrases		
<input type="checkbox"/> Training phrases	# words	
<input type="checkbox"/> Quiero tomar un medicamento	4	
<input type="checkbox"/> Estoy pensando en comer un alimento, me puedes ayudar?	9	
<input type="checkbox"/> preventivo	1	
<input type="checkbox"/> ¿Puedes asistirme con medicamentos?	4	
<input type="checkbox"/> necesito información sobre medicamentos	4	
<input type="checkbox"/> Oye quiero hacer una pregunta medicamentos.	6	
<input type="checkbox"/> necesito ayuda	2	

Parameter id	Entity type	Is list	Redact in log
medicamento	@medicamento	<input type="checkbox"/>	<input type="checkbox"/>

Figura 6 Frases de entrenamiento para el modo preventivo.

<input type="checkbox"/> Training phrases	# words	
<input type="checkbox"/> informativo	1	
<input type="checkbox"/> Estoy muy mal.	3	
<input type="checkbox"/> No se que me pasa, me duele el cuerpo.	9	
<input type="checkbox"/> Me duele la cabeza.	4	
<input type="checkbox"/> Me siento mal.	3	
<input type="checkbox"/> Soy diabético y siento un dolor punzante en el abdomen.	10	
<input type="checkbox"/> Tengo diabetes y dificultad para respirar.	6	
<input type="checkbox"/> Soy un adulto mayor con diabetes y tengo dolor de espalda.	11	
<input type="checkbox"/> Me duele una muela, y soy diabético.	7	
<input type="checkbox"/> Tengo diabetes y me pica todo el cuerpo.	8	

Items per page: 10 ▾ 1 - 10 of 20 < > << >>

- Clasificación de intenciones.

La clasificación de intenciones es un componente clave del Reconocimiento de Entrada, ya que ayuda a comprender las intenciones detrás de las solicitudes de los usuarios. Al utilizar todos los procesos descritos en conjunto, el agente es capaz de ofrecer interacciones significativas y de alta calidad, brindando respuestas precisas y relevantes a las necesidades y objetivos de los usuarios.

3.2.2. Comprensión del lenguaje

Los elementos del componente Comprensión del Lenguaje, tienen una función esencial en el procesamiento eficiente del lenguaje natural dentro del sistema de diálogo. Para cumplir esta función hace uso de diferentes elementos que son descritos a continuación.

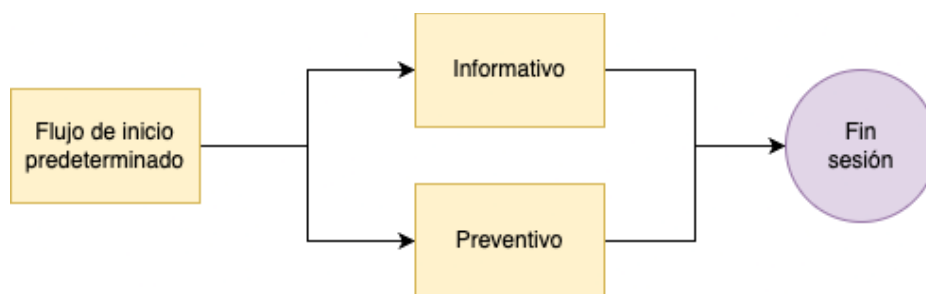
- Flujos

Los flujos son herramientas esenciales que nos permiten estructurar temas y las rutas de conversación vinculadas a ellos. Cada agente dispone de un flujo preestablecido denominado 'Flujo de inicio'. Este único flujo puede ser suficiente para el funcionamiento de un agente de menor complejidad (Google Cloud, s.f.).

Una conversación suele incluir varios tópicos. En el caso del agente propuesto, se plantean dos temas esenciales; preventivo e informativo. Cada uno de ellos forma un escenario

diferente. El escenario preventivo, enfocado a recomendar al usuario si el medicamento o alimento que tiene en mente consumir, es adecuado para su condición, este escenario es representado por el flujo 'Preventivo'. El escenario informativo está enfocado a ayudar a identificar luego de que ya haya consumido un alimento o medicamento, si ese es el causante de algún malestar que padece el usuario en ese momento. Para poder obtener la información para estos dos escenarios, se necesita de varios turnos de conversación. Los flujos definidos se muestran en la Figura 7.

Figura 7 Definición de los flujos del agente conversacional para ejecutar recomendaciones.



- Páginas

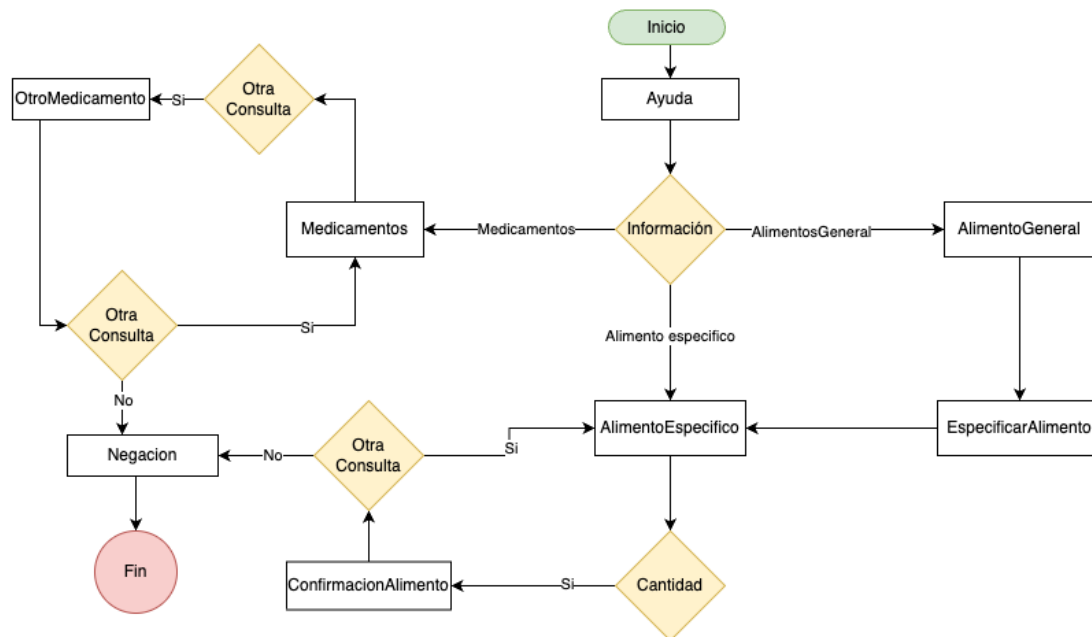
Las páginas tienen la capacidad de dirigir una conversación completa en torno a temas para los cuales se ha diseñado el flujo. En un momento dado, sólo una página es la actual, siendo esta la que se considera activa, y el flujo correspondiente a dicha página también se considera activo; cada flujo posee una página de inicio particular y al activarse, la página en curso se convierte en la página de inicio. (Google Cloud, s.f.).

Dentro del agente se han definido varias páginas, las cuales tienen la responsabilidad de obtener datos que permiten interactuar y continuar la conversación. A continuación, se proporciona una descripción detallada de las páginas que entran en juego en ambos escenarios:

- Escenario preventivo

En la Figura 8 se ilustran las páginas para extraer información sobre los alimentos que el usuario pretende consumir en el escenario preventivo.

Figura 8 Definición del escenario preventivo del agente conversacional.



El escenario preventivo se enfoca en la interacción con el usuario para recolectar información sobre alimentos y fármacos. Las páginas se irán habilitando de acuerdo a las respuestas proporcionadas por los usuarios. El proceso comienza en la página denominada 'Ayuda', que tiene como función principal determinar si el usuario requiere información sobre alimentos o fármacos. En este caso, se hace hincapié en la extracción de información sobre alimentos. Una vez identificada la preferencia del usuario por la información alimentaria, que se lo hace a través de intenciones, las cuales se detallan más adelante en el apartado 'Intenciones', surgen varias posibilidades. El usuario puede proporcionar detalles sobre un alimento en general, especificar un alimento en particular, o decidir que ya no necesita información adicional.

Si el usuario opta por proporcionar información sobre un alimento de manera general, la página 'Alimento General' se encarga de recolectar los datos pertinentes. Y le pide al usuario que ingrese un alimento específico.

En el caso de que el usuario ya haya especificado un alimento, la página 'Alimento Especifico' entra en acción. En este punto, pueden surgir dos alternativas: el usuario especifica la cantidad del alimento, pasando directamente a la página 'Confirmacion Alimento', o no proporciona dicha información, activando la función de la página 'Cantidad'. Esta última página está diseñada para obtener detalles sobre la cantidad del alimento, ya sea en gramos o en cantidades enteras (por ejemplo, una manzana, dos peras, etc.).

Una vez recolectada la información sobre el alimento, se consulta al usuario si desea obtener información sobre otro alimento, tarea que recae en la página 'Confirmacion Alimento'. En caso de una respuesta negativa, el flujo de interacción conduce al usuario a la página 'Negación', la cual posee una funcionalidad bidireccional, permitiendo retornar a la página inicial 'Ayuda' si el usuario decide cambiar su enfoque hacia la consulta de medicamentos. Si el usuario decide no buscar más información, la interacción culminaría en la página 'Fin de sesión'.

➤ Escenario Informativo

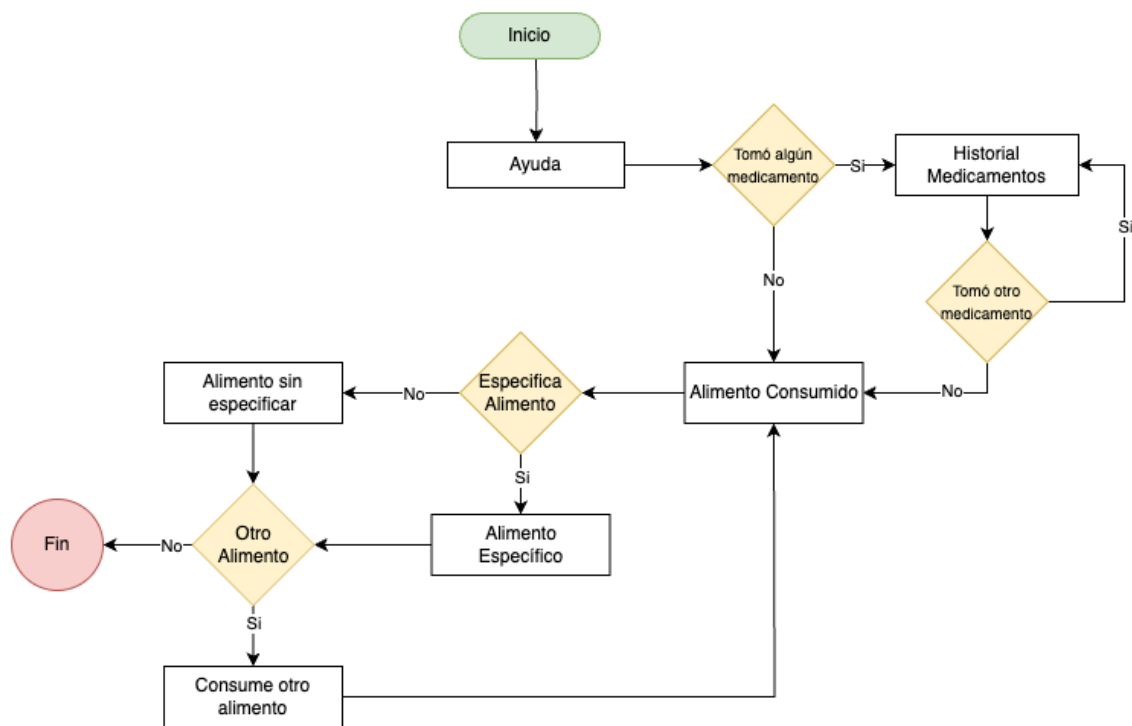
Por otra parte, el escenario informativo está enfocado en situaciones donde el usuario ha ingerido un medicamento que no forma parte de su régimen habitual o ha consumido un alimento que potencialmente ha originado una interacción (Figura 9). Se busca evaluar la ocurrencia de estas dos situaciones. La primera radica en el interés del usuario a entender el motivo detrás de su estado actual de malestar. Este flujo de información inicia con la página 'Start' y es representada con la frase 'me siento mal'.

Posteriormente, se procede a la página 'Ayuda', que tiene la tarea de recopilar la información sobre los medicamentos que el usuario ha estado tomando regularmente. La siguiente etapa involucra la página 'Medicamento consumido', que busca determinar si el usuario ha ingerido algún medicamento fuera de su régimen establecido. Si la consulta revela que el usuario ha consumido otro medicamento, se le proporciona información sobre las posibles interacciones, si existen.

En caso de no detectar alguna interacción entre medicamentos, se avanza hacia la segunda posible situación basada en la exploración de la ingesta de alimentos por parte del usuario y a través de la página 'Alimento Consumido'. Si el usuario menciona un tipo de alimento de forma genérica, la página 'Alimento sin especificar' solicita que se proporcione una descripción más detallada del alimento. De lo contrario, la página 'Alimento específico' se encarga de investigar si la interacción provino de la ingesta de ese alimento.

Además, la página 'Consumir otro alimento' funciona como un bucle que permite al sistema solicitar información sobre varios alimentos y compararlos con los medicamentos. En caso de que no se identifique ninguna interacción, se concluye la sesión y se insta al usuario a que siempre consulte con un experto en caso de malestar.

Figura 9 Definición del escenario informativo del agente conversacional.



- Tipos de Entidades

Los tipos de entidades desempeñan una función en la gestión de cómo se extraen los datos a partir de la entrada proporcionada por el usuario. La mayoría de plataformas de comprensión del lenguaje natural ofrecen una gama de entidades predefinidas capaces de concordar con una amplia variedad de datos comunes. Ejemplos de esto son entidades que pueden reconocer fechas, horas, colores, direcciones de correo electrónico, entre otros. No obstante, también se brinda la posibilidad de crear entidades personalizadas que se adecuen a la detección de concordancias en datos específicos (Google Cloud, s.f.). En el desarrollo del agente se crearon entidades para los alimentos y los medicamentos. En la tabla adjunta se muestra un ejemplo de los tipos de entidades creadas para mejorar la comprensión del lenguaje.

Tabla 1 Definición de la entidad alimentos.

Alimentos	
Entidad	Términos semejantes
Lácteos	Leche entera, Helado de crema, Queso,
Frutas	Manzana, Mango, Toronja,

La tabla que se presenta a continuación proporciona una clasificación de las diversas formas en que el usuario puede ingresar información durante la interacción, ya sea a través de la entidad directa o de algún término semejante previamente definido. El agente implementado en Dialogflow CX posee la capacidad de inferir qué entidad ha sido descrita por el usuario, o identificar el valor original que se introdujo. Esta funcionalidad permite abordar el análisis desde dos perspectivas distintas, adaptándose a las necesidades particulares del escenario en cuestión. Además, para abordar el tema de los medicamentos, se definieron las entidades como se muestra la Tabla 2.

Tabla 2 Definición de la entidad de medicamentos.

Medicamentos	
Entidad	Términos semejantes
Abatacepto	Orencia, abatacepto, abatasepto, abataseptp, avatacepto
Metformina	Metformina

La sección de términos semejantes hace referencia a la marca comercial, fabricante o las diversas denominaciones con las que puede identificarse un medicamento. En este sentido, se han incorporado variantes con gramática incorrecta. La razón de esta inclusión radica en la manera en la que el usuario interactúa con el aplicativo móvil, la cual se realiza a través de comandos de voz.

Al convertir la entrada de audio en texto, el sistema, en ocasiones, interpreta las palabras de forma aproximada, especialmente si estas corresponden a medicamentos, por tal motivo, se han definido términos semejantes con estas variaciones para garantizar la correspondencia adecuada. Esta estrategia garantiza mayor flexibilidad y adaptabilidad en la interacción,

optimizando la comprensión del sistema ante posibles interpretaciones en la entrada de los usuarios.

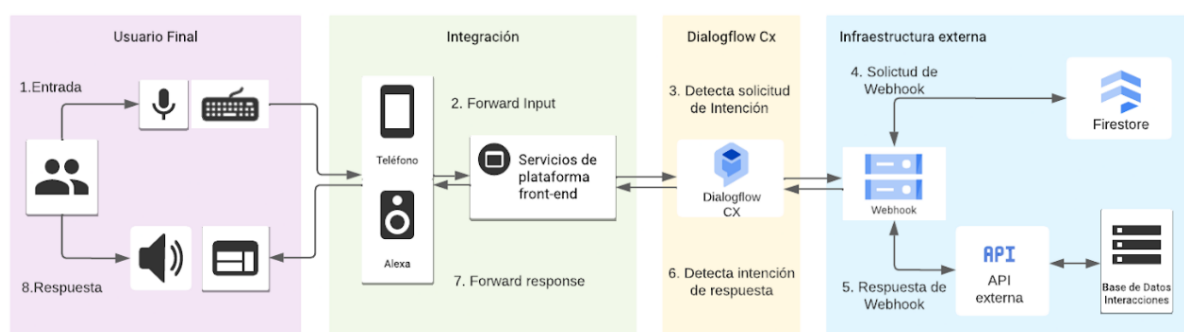
- Parámetros

Este componente cumple una función esencial al permitir la captura y referencia de los valores finales introducidos por el usuario durante una sesión. Los 'valores finales' se refieren a los datos específicos, como el tipo de alimento consumido, el medicamento tomado y las porciones ingeridas. Cada uno de estos parámetros se caracteriza por un nombre y un tipo de entidad (Google Cloud, s.f.). En este proyecto, los parámetros, al ser representaciones de datos estructurados, se encuentran definidos en cada página y son empleados para almacenar información, la cual es transmitida al webhook para su procesamiento de acuerdo con las solicitudes del usuario.

3.2.3. Generador de respuestas

Este componente se encarga de los mensajes que serán generados al usuario, los cuales se producen mediante *Interacciones*. Las interacciones corresponden a las solicitudes del usuario y a las respuestas que son procesadas por el agente dentro del flujo de conversación. En el transcurso de cada interacción, el usuario realiza una solicitud, la cual es procesada por el agente conversacional, que a su vez genera y envía una respuesta. A continuación, se detallan los pasos para la comunicación entre el agente y el usuario (Figura 10):

Figura 10 Flujo del agente conversacional.



Nota: Flujo del agente conversacional, adaptado de (Google Cloud, s.f.).

1. La interacción del usuario final está diseñada para realizarse tanto mediante la voz como por texto. Es relevante mencionar que, debido a limitaciones técnicas específicas de ciertas plataformas, como Android, la funcionalidad de voz puede presentar desafíos, un aspecto que se abordará en mayor detalle en la sección Limitaciones del estudio.

2. Se ha centrado en el desarrollo de una aplicación móvil que permite a los usuarios realizar consultas a través de un chat. Como un aspecto adicional pero relevante, se ha añadido una compatibilidad básica con Alexa. Aunque esta integración no es el núcleo del presente trabajo, se ha incluido esta capacidad de interacción para sentar las bases de futuras mejoras. Esta compatibilidad se logra mediante el SDK de Alexa, que posibilita la creación de una ‘habilidad’ (Skill) para Alexa, permitiendo así una interacción básica con el agente.
3. Al recibir la solicitud del usuario final, se envía al agente conversacional, que, a través de las intenciones, identifica la solicitud del usuario.
4. Tras el análisis de la solicitud, esta se envía a un webhook, responsable de la conexión con servicios externos. En primera instancia, se dirige a una base de datos, encargada de almacenar la información del usuario. Este extrae la información relevante, ya sea sobre alimentos o medicamentos, y posteriormente la envía a la API. La API interactúa con la base de datos de interacciones, analiza la solicitud y busca cualquier interacción pertinente relacionada con los alimentos o medicamentos.
5. Tras la consulta en la base de datos de interacciones, el sistema retorna una respuesta sobre las interacciones al sistema mediante el webhook.
6. Posteriormente, se detecta la intención de la respuesta y esta se envía al front-end.
7. El front-end es el encargado de presentar la información ya sea a través de la aplicación móvil o de Alexa.
8. La presentación al usuario final varía dependiendo del método utilizado. Si el usuario emplea la aplicación móvil, podrá visualizar y escuchar la respuesta del agente. Sin embargo, si el usuario opta por Alexa, solo podrá escuchar la respuesta.

3.2.4. Renderizado

Este componente se encarga de realizar el renderizado a través de texto. Sin embargo, para la transformación de texto a voz, es la Aplicación Móvil la responsable de esta funcionalidad. La aplicación, en su papel de interfaz directa con el usuario, desempeña el renderizado a voz. Esta operación se materializa gracias a su enlace con la *API de la Interfaz de Usuario*, lo cual facilita la continuidad de la interacción en el agente conversacional y estimula una comunicación fluida y natural. Más detalles sobre el diseño e implementación de la Aplicación Móvil pueden ser consultados en el Capítulo 5.

3.3. Dominio específico de componentes

Este elemento del sistema proporciona los detalles del dominio con los que interactúa el *Agente Conversacional*. El *Extractor de Información Específica* extrae los datos necesarios, que serán usados para verificar las interacciones medicamentosas. Luego, el *Webhook* se encarga de la orquestación del proceso y finalmente el *Algoritmo de recomendación* se encarga de dar una alternativa en caso de existir una interacción fármaco-alimento. A continuación, se detalla cada uno de los componentes en orden de importancia.

3.3.1. Extractor de información específica

Un elemento esencial en la configuración de este sistema es el *Extractor de Información Específica*. Este componente se encarga de la administración y manipulación de la información requerida para generar recomendaciones personalizadas. Para garantizar un almacenamiento seguro y escalable, se ha optado por las plataformas Firebase y Supabase.

Firebase, gracias a sus servicios en tiempo real y una integración estrecha con la aplicación móvil, posibilita un manejo eficaz de los datos. Por otra parte, Supabase se ha seleccionado para gestionar la lógica de interacciones de fármacos y alimentos a través de una base de datos relacional, permitiendo así manejar relaciones complejas y realizar consultas más sofisticadas. Esta elección de infraestructura de almacenamiento de datos es crítica para garantizar la calidad y precisión de las recomendaciones proporcionadas por nuestro sistema.

- **Firestore**

En el desarrollo de este proyecto, la plataforma Firestore permite la gestión de la persistencia de la sesión del usuario, así como el almacenamiento de datos personales: nombre, apellido, edad, género, teléfono, horarios de alimentación y el registro de medicamentos que consume.

Además de esto, Firestore aporta un valor añadido gracias a sus servicios en tiempo real. Estos servicios permiten generar notificaciones a los usuarios basadas en sus horarios de alimentación. Cuando la hora de una comida desayuno, almuerzo o merienda está próxima, se envía una notificación al usuario. Esta alerta temprana incentiva a que el usuario consulte en la aplicación los alimentos que planea consumir en esa comida para evitar alguna interacción, un aspecto que se explicará con mayor detalle en las siguientes secciones.

Un aspecto importante a resaltar es la estructura de almacenamiento de datos en Firestore, la cual utiliza una organización jerárquica de documentos y colecciones. En nuestro caso, utilizamos la colección 'Usuarios', la cual alberga a todos los individuos registrados en la aplicación. Cada usuario posee un documento correspondiente dentro de la colección. Estos documentos están vinculados a identificadores únicos generados por el sistema de

autenticación de Firebase para cada usuario. De este modo se asegura la existencia de un identificador singular y consistente para cada usuario.

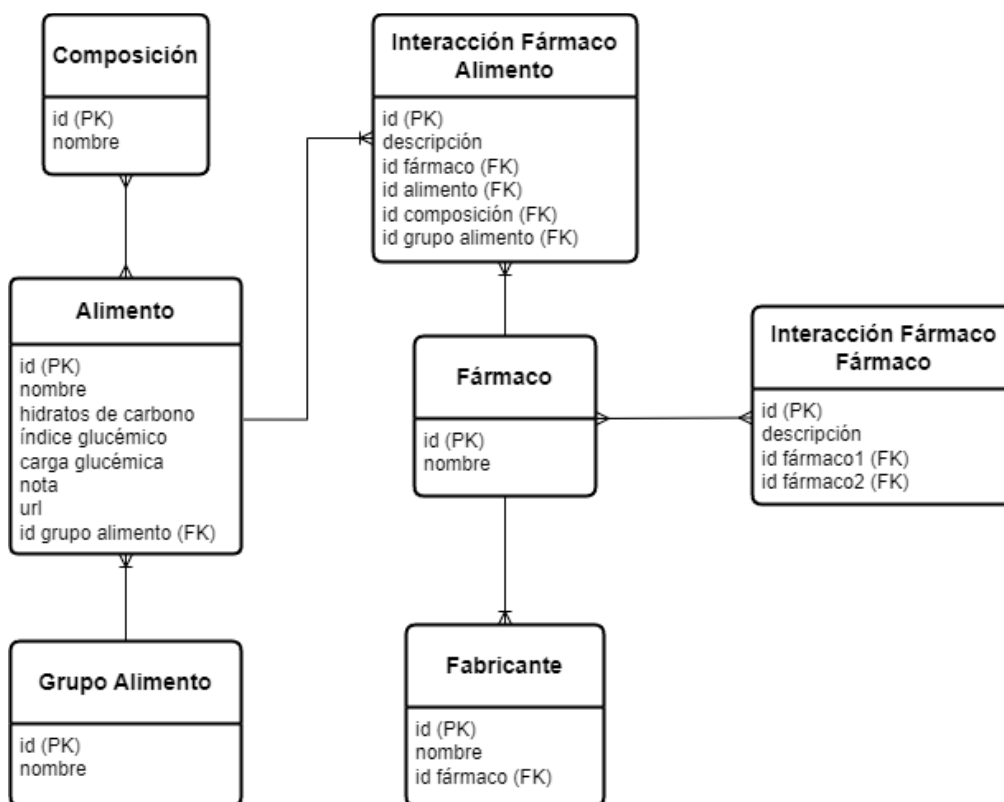
A continuación, se describe la implementación de los atributos definidos en Firebase:

- Usuario: hace referencia a la estructura de datos de tipo mapa⁴ que almacena la información personal del usuario. Los atributos vinculados a esta estructura son:
 - Correo
 - Nombre
 - Fecha de nacimiento
 - Medicamentos
 - Horarios de Alimentación: Registra los horarios de las comidas del usuario; puede estructurarse como una estructura de datos de tipo mapa. Los atributos asociados pueden incluir:
 - Desayuno: Almacena la hora programada para el desayuno
 - Almuerzo: Almacena la hora programada para el almuerzo
 - Merienda: Almacena la hora programada para la merienda, si corresponde
 - Medicamentos: Almacena los medicamentos que el usuario toma regularmente; estructurado como una estructura de datos de tipo arreglo.
- Supabase

Supabase permite diseñar y gestionar las entidades restantes que son necesarias para la construcción del sistema. Las tablas contienen información sobre alimentos, composición nutricional, medicamentos y posibles interacciones fármaco-fármaco y fármaco-alimento (Figura 11). Mediante el uso de Supabase, es posible almacenar y consultar fácilmente estos datos.

⁴El objeto Mapa es una estructura de datos que almacena pares de clave-valor, manteniendo el orden original de inserción de las claves (Mozilla Developers, 2023)

Figura 11 Modelo de base de datos sobre alimentos, fármacos e interacciones.



Cabe destacar que Supabase no es estática, sino que su contenido es susceptible de ser actualizado y ampliado. Esto es posible gracias a la API (sección 3.4) que permite la incorporación de nuevos medicamentos e interacciones fármaco-fármaco. Sin embargo, es crucial mencionar que estas actualizaciones y adiciones son realizadas por un administrador, asegurando así un manejo cuidadoso y preciso de la información. Este diseño no solo facilita el mantenimiento del sistema, sino que también garantiza que la información proporcionada al usuario sea siempre relevante y actual.

A continuación, se detallan cada una de las tablas que conforman la base de datos, con el objetivo de ofrecer una comprensión clara y precisa del contexto en el que se encuentra el sistema de recomendación.

➤ Alimento

Almacena información relevante acerca de los alimentos, tales como su nombre, contenido de carbohidratos, índice glucémico, carga glucémica, notas adicionales, URL asociada y el grupo de alimentos al que pertenecen; consta de 243 registros. Esta tabla resulta de gran utilidad para organizar y almacenar datos relacionados con diferentes tipos de alimentos

Los campos de esta tabla son:

- Id (clave primaria): Representa el identificador único de cada registro del alimento en la tabla.
- Nombre: Indica el nombre del alimento.
- hidratos de carbono: Se refiere a la cantidad de carbohidratos presentes en el alimento⁵.
- Índice glucémico: Es un valor numérico que indica el impacto del alimento en los niveles de glucosa en sangre⁵.
- Carga glucémica: Representa la carga glucémica del alimento⁵.
- Nota: Proporciona una nota o descripción adicional sobre el alimento, sirve para colocar información referente al alimento⁵.
- Url: Es una URL o enlace asociado al alimento, contiene la imagen del mismo.
- Id grupo alimento (clave foránea): hace referencia al identificador del grupo de alimentos al que pertenece el alimento. Los grupos de alimentos se han considerado de una fuente de datos⁵.

Los conceptos de hidratos de carbono, índice glucémico y carga glucémica se abordarán con más detalle en la sección 3.3.

➤ Composición

Almacena datos relacionados a la composición nutricional o ingredientes utilizados en los alimentos. Proporciona una manera de catalogar y organizar los componentes individuales que se encuentran en los alimentos; permiten evaluar el contenido nutricional de los alimentos y proporcionar recomendaciones acordes a las necesidades específicas del usuario. Los campos que componen esta tabla se especifican en la Figura 12.

Figura 12 Descripción de almacenamiento de datos en la tabla composición.

id int4	nombre text
1	Calcio
2	Potasio
3	Magnesio
4	Vitamina B
5	Hierro

⁵Datos considerados respecto a las tablas de nutrición y dietética (Meneses, 2020)

➤ Grupo Alimento

Representa información detallada sobre los distintos grupos de alimentos. Cada grupo cuenta con un registro que incluye su nombre o descripción correspondiente. Esta clasificación y organización facilita la gestión y búsqueda de información relacionada con los grupos de alimentos.

En este contexto, se han establecido ocho grupos principales basados en la pirámide alimenticia, tales como: lácteos, verduras, frutas, frutos secos, cereales y tubérculos, legumbres, proteínas y alcohol (Díaz et al., 2006). Los atributos de esta tabla están representados en la Figura 13.

Figura 13 Descripción de almacenamiento de datos en la tabla grupo alimento.

id int2	nombre text
1	Lacteos
2	Verduras
3	Frutas
4	Frutos secos
5	Cereales y Tuberculos
6	Legumbres
7	Proteinas
8	Alcohol

➤ Fármaco

Contiene información sobre diversos medicamentos, es utilizada para organizar y almacenar datos relacionados con los fármacos utilizados en la práctica médica relacionado con la diabetes. Mismo que se almacena de la siguiente manera.

Figura 14 Descripción de almacenamiento de datos en la tabla Fármaco.

id int4	name varchar
1	Metformina

➤ Fabricante

Almacena información sobre los fabricantes o laboratorios farmacéuticos que producen los fármacos o medicamentos registrados en la tabla 'Fármaco'. Esta información es utilizada para definir los diferentes nombres con los que se conoce a un medicamento. Por ejemplo, los componentes de la Figura 14, pertenecen a un medicamento definido mediante el id, que en este caso es la Metformina (Figura 15).

Figura 15 Descripción de almacenamiento de datos en la tabla Fabricante.

id int4	nombre varchar	id_farmaco int4
1	Actoplus Met	1
2	Avandamet	1
3	Fortamet	1

➤ Interacción fármaco-alimento

Comprende información relevante sobre las posibles interacciones entre fármacos y alimentos, proporcionando detalles específicos sobre la descripción de la interacción, el alimento, composición y grupo de alimento al cual pertenece. En cada registro se toma en cuenta únicamente uno de los tres componentes mencionados para las posibles interacciones (Figura 16).

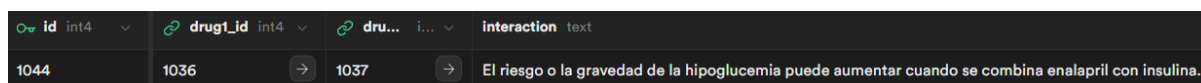
Figura 16 Descripción de almacenamiento de datos en la tabla Interacción fármaco-alimento.

id int4	descripcion text	id_far... i...	id_alim... i...	id_composi... i...	id_grupo_alime... i...
3	Evite los productos que contengan potasio,	1037	NULL	2	NULL
5	Evite el alcohol.	1	243	NULL	NULL
1	Evite los productos que contengan toronja; l	989	207	NULL	NULL
6	Evite los productos que contengan toronja; l	989	242	NULL	NULL

➤ Interacción fármaco-fármaco

Esta tabla tiene como finalidad el registro de las posibles interacciones entre diferentes medicamentos, estableciendo una relación entre dos fármacos en cada registro y proporcionando detalles sobre su interacción, que puede incluir posibles efectos adversos. La información contenida es de gran relevancia para el sistema de recomendación, ya que permite la identificación de combinaciones peligrosas o contraindicaciones al momento de tomar múltiples medicamentos simultáneamente. Un ejemplo de la información contenida en esta tabla se muestra en la Figura 17.

Figura 17 Descripción de almacenamiento de datos en la tabla interacción fármaco fármaco.



id	drug_id	dru...	interaction
1044	1036	1037	El riesgo o la gravedad de la hipoglucemia puede aumentar cuando se combina enalapril con insulina.

Estas tablas permiten el registro y análisis de información relevante acerca de los alimentos, su composición nutricional, los medicamentos, los fabricantes y las posibles interacciones entre alimentos y medicamentos. La utilización de la base de datos ayuda al sistema de recomendación a generar sugerencias personalizadas para cada usuario, teniendo en consideración sus preferencias alimentarias, necesidades médicas y posibles interacciones. De esta manera proporciona una experiencia más segura en la gestión de la alimentación y medicación; su descripción detallada resulta fundamental para comprender la estructura y funcionamiento de la base de datos en conjunto y comprender el valor que esta ofrece en el ámbito de la nutrición y la medicación personalizada.

- Webhook

Webhook⁶ permite una comunicación asíncrona y gestionada por eventos (event-driven) con el Agente conversacional. Esto significa que el webhook solo es invocado cuando es necesario, es decir, cuando se generan eventos relevantes en la conversación. Esto reduce la carga de procesamiento y optimiza el rendimiento general del sistema.

Configurar un webhook en Dialogflow CX es relativamente más sencillo en comparación con establecer una conexión WebSocket⁷. El uso de Webhook requiere proporcionar una URL donde Dialogflow CX pueda enviar las solicitudes HTTP, mientras que el uso de WebSockets implica una configuración adicional del servidor y la implementación de la lógica de manejo de eventos en ambos extremos. Entre las responsabilidades del Webhook están:

- Manejo de solicitudes

Webhook, en su rol de intermediario, recibe peticiones procedentes del agente implementado en Dialogflow CX con el propósito de adquirir recomendaciones sobre medicamentos. Dichas peticiones pueden contener detalles como el nombre del fármaco que el usuario planea consumir, además de los alimentos que pretende ingerir junto con la cantidad correspondiente.

- Interacción con el agente conversacional

La comunicación entre Webhook y el agente conversacional implementado con Dialogflow es bidireccional.

⁶Un webhook es una función de callback basada en HTTP que permite la comunicación entre dos APIs de forma liviana y basada en eventos. (RedHat, 2023).

⁷WebSocket es un protocolo estándar que permite una aplicación cliente y una aplicación de servidor web utilice una conexión dúplex para comunicarse (IBM, 2023).

Webhook recibe las solicitudes formuladas por el agente de Dialogflow y tras el procesamiento de las mismas, remite las respuestas al agente para comunicarlás al usuario.

➤ Elaboración de respuestas

Webhook, tras recibir las solicitudes del agente conversacional, se encarga de procesar y generar las recomendaciones que serán enviadas a la interfaz del usuario. Durante este proceso, es plausible realizar un análisis para detectar posibles contraindicaciones o interacciones entre los medicamentos que el usuario planea consumir y aquellos que ya forman parte de su ingesta habitual. En pocas palabras, webhook es el orquestador de todo el proceso del agente conversacional. A continuación, se detalla el procedimiento para la comprobación de Interacciones Medicamentosas:

- Proceso de comprobación de interacciones medicamentosas

El *Proceso de Comprobación de Interacciones Medicamentosas* es el componente central de este trabajo y se centra en consultar y verificar las interacciones entre fármacos y entre fármacos y alimentos. En el caso de las interacciones fármaco-fármaco, el objetivo del procedimiento es determinar si existe alguna interacción entre un medicamento que el usuario pretende consumir y los medicamentos que ya se encuentran en su régimen de tratamiento. En cuanto a las interacciones fármaco-alimento, se verifica si hay alguna interacción entre el alimento que el usuario planea consumir y los fármacos que está tomando. Es crucial para la efectividad de este proceso que sea capaz de identificar correctamente estos parámetros y dar a conocer al usuario las interacciones que existen. La estructura del algoritmo está diseñada para manejar dos tipos de escenarios: informativo y preventivo. Se detallan los elementos principales del algoritmo.

- Tareas del proceso

El proceso de comprobación se basa en actividades como la recolección y procesamiento de la información, análisis de interacciones y generación de alertas, y los parámetros utilizados para realizarla. Es importante mencionar que las actividades trabajan de manera conjunta y se desglosan de la siguiente manera:

▪ Recolección y procesamiento de información

La adquisición inicial de datos se destaca como la primera actividad del proceso de comprobación propuesto. Esta información es primordial para el funcionamiento óptimo del sistema y se recopila a través del agente conversacional (componente descrito previamente). La información se recopila principalmente en forma de lenguaje natural escrito mediante la interfaz de la aplicación móvil desarrollada, aunque también se ha diseñado una opción que admite entradas de voz en ciertos dispositivos. Sin embargo, es relevante mencionar que la

funcionalidad de voz puede presentar limitaciones en algunas plataformas, un aspecto que se abordará más adelante en la sección 5.2.

En el caso del escenario preventivo, los parámetros clave recopilados son los alimentos y medicamentos que el usuario planea consumir. Dado que el agente está diseñado para ofrecer información personalizada, recopila los datos sobre cada medicamento almacenado por el usuario. Esta información resulta crucial para identificar posibles interacciones entre los medicamentos del usuario y lo que está a punto de consumir. Aunque los alimentos no se almacenan de manera personalizada, se registran en tiempo real cuando el usuario los ingresa a través de la aplicación.

Para complementar el funcionamiento del proceso de comprobación de interacciones, es pertinente enfocar nuestra atención en el escenario informativo. En este caso, la recopilación de parámetros se centra en lo que el usuario ya ha consumido. En primera instancia, se recolecta información referente al medicamento que ha ingerido recientemente con el objetivo de investigar potenciales interacciones con otros fármacos que el usuario ha estado consumiendo. Si esta primera búsqueda no resulta en hallazgos significativos, el proceso prosigue con la recopilación de datos sobre los alimentos que el usuario pudo haber ingerido recientemente. Esta etapa de la búsqueda se enfoca en determinar si estos alimentos pudieron ser el desencadenante de una interacción fármaco-alimento. De esta manera, se realiza una exploración exhaustiva y sistemática para identificar los posibles factores de interacción que afectan al usuario.

- Análisis de interacciones

La segunda actividad fundamental del proceso de comprobación propuesto se enfoca en el análisis de las interacciones. Los elementos vitales para este proceso son los medicamentos del usuario, tanto los que se consumen habitualmente como los que se planea consumir, así como la ingesta de alimentos.

En el escenario preventivo, una vez recopilada esta información, el sistema realiza un análisis exhaustivo, comparando cada uno de los medicamentos proporcionados por el usuario con la lista de alimentos que planea consumir. Para este fin, el sistema consulta la base de datos de Supabase, evalúa la información y devuelve cualquier interacción detectada, si es que existe.

En el escenario informativo, se recopila y analiza información tanto de los medicamentos consumidos recientemente como de los alimentos ingeridos por el usuario. El objetivo es determinar si los síntomas experimentados por el usuario pueden ser el resultado de una interacción farmacológica o de una interacción entre un medicamento y un alimento. Si se

detecta una interacción, el sistema informa al usuario y proporciona información adicional sobre la naturaleza de dicha interacción. Para detectar las posibles interacciones con respecto al malestar que siente, el algoritmo procede a realizar preguntas exhaustivas sobre los medicamentos que el usuario ha consumido recientemente y sobre los alimentos que ha ingerido. A diferencia de este, el escenario preventivo se basa principalmente en la anticipación, ya que busca identificar potenciales interacciones basándose en lo que el usuario planea consumir, tanto en términos de medicamentos como de alimentos.

- Generación de alertas

La actividad de generación de alertas por interacciones es el tercer aspecto esencial del proceso. Este se lleva a cabo mediante el Webhook del agente, que tiene la tarea de proporcionar información relevante basada en el análisis de las interacciones registradas.

Para los casos preventivos e informativos de interacciones fármaco-fármaco, se generan orientaciones específicas que el Agente Conversacional comunica al usuario, indicando si se ha identificado una interacción y especificando su tipo. Si no se detecta ninguna interacción, la respuesta correspondiente se emite.

En el caso de las interacciones fármaco-alimento, se realizan dos tipos de evaluaciones: una verificación de la interacción fármaco-alimento y una valoración de la conveniencia del alimento para el usuario. La primera verificación se enfoca en identificar posibles interacciones del alimento con cualquier medicamento que el usuario pueda estar consumiendo.

Este proceso se ejecuta de tres maneras. Primero, se verifica directamente la interacción del alimento con el medicamento. Un ejemplo de esto se encuentra en los hallazgos de Drier (2020), quien sostiene que ciertos alimentos, como las naranjas, los plátanos y las verduras de hojas verdes, pueden interactuar con inhibidores de la ECA, como Enalapril y Captopril. Esto abarca el caso de la interacción directa con el alimento.

Si no se encuentra ninguna interacción en este nivel, el proceso avanza hacia la verificación de interacciones con el grupo alimenticio al que pertenece el alimento en cuestión. Este paso se realiza debido a que algunos medicamentos pueden interactuar con grupos alimenticios específicos. Si no se detecta ninguna interacción en este nivel, se verifica en función de los componentes nutritivos del alimento. Drier (2020) afirma que el consumo excesivo de ciertos nutrientes puede interactuar de manera adversa con algunos medicamentos, lo que justifica este nivel de verificación.

Si se detecta alguna interacción, el Webhook devuelve al Agente la descripción de dicha interacción y el medicamento implicado. Si no se identifica ninguna interacción, se procede con la segunda verificación que analiza la carga glucémica del alimento, concepto que se detalla en la siguiente temática.

- Parámetros del proceso de comprobación de interacciones medicamentosas

En esta sección, se describen los parámetros clave que son esenciales para garantizar un funcionamiento efectivo en la identificación de interacciones fármaco-alimento. En situaciones en las que no se detecta una interacción, nuestro enfoque se expande más allá del análisis básico y busca proporcionar información adicional al usuario con base en la carga glucémica de los alimentos. Este enfoque ampliado está diseñado para mejorar la funcionalidad del algoritmo y proporcionar un beneficio adicional, aportando una perspectiva adicional a la hora de tomar decisiones sobre la dieta.

La evaluación de la carga glucémica es un componente clave de este enfoque expandido y para calcularla es necesario conocer tanto el índice glucémico como la cantidad de hidratos de carbono en el alimento en cuestión. Estos aspectos se detallan a continuación.

Cabe destacar que este enfoque de análisis ampliado no se limita a la identificación de interacciones fármaco-alimento, sino que también busca proporcionar un nivel adicional de información y orientación sobre la dieta, incluso en ausencia de interacciones detectadas.

- Índice glucémico

De acuerdo con Lo y Abihm (2018) el índice glucémico (IG) evalúa el incremento de glucosa e insulina en la sangre ocasionado por el consumo de un alimento específico. Por su parte, Kim (2020) sostiene que el IG es uno de los múltiples valores empleados para medir la velocidad en la que el cuerpo digiere o descompone los carbohidratos.

Los alimentos se califican en una escala de 0 a 100, donde la glucosa pura (azúcar) posee un valor de 100. Un alimento con un índice glucémico más bajo incrementa el azúcar en la sangre a un ritmo más lento tras su ingesta. En términos generales, cuanto más procesado es un alimento, mayor será su IG, y cuanto más fibra contenga, menor será su IG (Harvard Medical School, 2021).

- Hidratos de carbono

Este parámetro hace referencia a la fracción de carbohidrato que, dentro de los alimentos, puede ser procesada, asimilada y metabolizada en calidad de carbohidrato; ocasionalmente también se le identifica como carbohidrato neto o glucémico (Augustin et al., 2015). La ingesta

de hidratos de carbono puede influir en la elevación de los valores de glucosa en el torrente sanguíneo (Eleazu, 2016).

Es importante aclarar que, a pesar de que los hidratos de carbono pueden influir en la glucosa sanguínea, Lo y Abihm (2018) sugieren que la asociación entre el consumo de hidratos de carbono y el aumento de los niveles de azúcar en la sangre es una representación simplista de la realidad. Esta asociación no siempre es predecible, debido a que las fluctuaciones en la concentración de glucosa e insulina en la sangre, luego de ingerir carbohidratos complejos, pueden presentar diferencias significativas. Además, Lu y Yan (2014) indican que los diversos tipos de carbohidratos, debido a sus propiedades físicas y químicas específicas, pueden dar lugar a respuestas variadas en la glucosa plasmática.

Sin embargo, el análisis de los hidratos de carbono en los alimentos tiene un propósito más allá de la simple identificación de su potencial para aumentar los niveles de glucosa en sangre. Los hidratos de carbono son un componente esencial para calcular la carga glucémica de un alimento. Esta medición es de gran utilidad en nuestro algoritmo para ofrecer al usuario una perspectiva más completa y contextualizada de su dieta.

- Carga glucémica

Según Lu y Yan (2014) la carga glucémica (CG) representa un método más completo y reciente para evaluar el impacto del consumo de carbohidratos, ya que tiene en cuenta tanto el IG como los hidratos de carbono presentes en una porción específica de un alimento. Mientras que el valor del IG indica la rapidez con la que un carbohidrato se convierte en azúcar, no proporciona información sobre la cantidad precisa de carbohidratos en ese alimento. Por lo tanto, para comprender el efecto de un alimento en los niveles de azúcar en la sangre, es necesario considerar ambos parámetros, y ahí es donde entra en juego la carga glucémica.

La CG se calcula sumando el producto del IG de cada alimento por la cantidad de hidratos de carbono y dividiendo el resultado por 100. En consecuencia, representa tanto la cantidad como la calidad de los carbohidratos en la dieta general, así como su interacción. Según Kim (2020) el tamaño de la porción puede variar según factores dietéticos y culturales.

Por lo tanto, se espera que un alimento de CG más alta, tenga un mayor impacto en los niveles de azúcar en la sangre y en la respuesta de insulina por porción, en comparación con un alimento con CG más baja. Un valor de carga glucémica de 10 o menos se considera bajo, de 11 a 19 se considera medio, y 20 o más se considera alto (Tabla 3).

Tabla 3 Valores de índice glucémico y carga glucémica.

Los alimentos que presentan bajo índice glucémico suelen tener baja carga glucémica. Sin embargo, aquellos alimentos con un índice glucémico intermedio o alto pueden tener un rango de carga glucémica muy amplio, desde muy bajo hasta muy alto. (Kim, 2020)

Índice glucémico	Carga glucémica
Si el número es más bajo, el alimento tendrá un menor efecto en el nivel de azúcar en la sangre.	Si el número es más bajo, la porción del alimento tendrá un menor efecto en el nivel de azúcar en la sangre.
55 o menos = bajo	10 o menos = bajo
56-69 = moderado	11-19 = moderado
70 o más = alto	20 o más = alto

Cabe señalar que este mecanismo se activa cuando no se detecta una interacción entre fármaco y alimento. Aunque el usuario final no recibe información explícita acerca de la carga glucémica, esta se traduce en tres categorías de mensajes: 'Puedes', cuando la carga glucémica está entre 0 y 9, 'Procura' cuando la carga glucémica se encuentra entre 10 y 19, y 'Alerta' cuando la carga glucémica supera el valor de 20, como se muestra en la tabla 3. Estas categorías se derivan del fragmento de código de la Figura 18, el cual refleja el criterio utilizado en la aplicación para categorizar los mensajes que se muestran al usuario, en función de la carga glucémica de los alimentos.

Figura 18 Categorías con respecto a la carga glucémica.

```

if (carga_glucemica < 10) {
    return `Puedes consumir ${nota}`;
} else if (carga_glucemica >= 10 && carga_glucemica < 20) {
    return `Procura consumir con cuidado ${nota}`;
} else {
    return `Alerta no consuma ${nota}`;
}

```

Es importante destacar que, al determinar la carga glucémica, también se tiene en cuenta el tamaño de las porciones. Por lo tanto, para identificar las diferentes formas de expresar un alimento (ya que, no es lo mismo decir 'un pan integral' que 'una rebanada de pan integral'), se ha desarrollado un modelo en el agente para detectar estas variaciones. Para ejemplificar, se establecieron algunos tipos, como se muestra en la Tabla 4, con el fin de definir valores específicos. Al definir estos valores, se obtiene un factor de conversión que se utiliza para calcular la carga glucémica cuando se trata de porciones de diferentes tamaños.

Tabla 4 Ejemplos de tipos de porciones de alimentos.

Tipo	Gramos
Unidad	100
Taza	240
rebanada	30
cucharada	15

El IG puede ser localizado en el sitio web oficial del índice glucémico y la base de datos internacional de IG, localizada en la Unidad de Nutrición Humana, Escuela de Biociencia Molecular, Universidad de Sydney, Grupo IG. Este grupo publica un boletín electrónico mensual que proporciona las investigaciones más recientes en relación a las indicaciones geográficas a nivel mundial (The University of Sydney, 2023).

En este sentido, el Centro Médico D-Médical de Madrid desarrolló información sobre los alimentos más consumidos, con el objetivo de ayudar a las personas con diabetes a controlar su dieta de manera efectiva (D-Médical, 2020). Utilizando dicha información, se modeló la tabla 'Alimento' en la base de datos, que incluye los parámetros de carga glucémica, índice glucémico e hidratos de carbono por cada 100 gramos, así como las descripciones relacionadas con su consumo recomendado. Dado que estas tablas proporcionan la información de los hidratos de carbono por cada 100 gramos, los factores de conversión brindan una perspectiva más amplia cuando el usuario ingresa porciones de diferentes tamaños. Esto permite tener una visión más generalizada de la carga glucémica de los alimentos.

Al disponer de estos parámetros (índice glucémico y carga glucémica), es posible utilizar la fórmula propuesta por Lu y Yan (2014):

Carga glucémica =

$$((\text{hidratos de carbono} \times \text{índice glucémico} \times \text{factor de conversión}) / 100) / 100$$

- Hidratos de carbono: Cantidad de hidratos de carbono presentes en el alimento; se puede obtener de la información nutricional del mismo.
- Índice glucémico: Medida para determinar el aumento de los niveles de azúcar en la sangre. Los alimentos se clasifican en una escala de 0 a 100, con la glucosa pura (azúcar) como referencia con un valor de 100.
- Factor de conversión: Se utiliza para convertir los hidratos de carbono en la cantidad de porciones que el usuario desea consumir, por ejemplo, una rebanada de pan, un pan, etc.

En esencia, la fórmula multiplica la cantidad de carbohidratos en gramos por el IG del alimento y el factor de conversión, luego divide el resultado entre 100 para obtener una versión normalizada. Posteriormente, este resultado se divide de nuevo entre 100 para transformarlo en un formato más comprensible y fácil de interpretar para los usuarios. El valor resultante brinda una estimación de cuánto un alimento específico puede alterar los niveles de glucosa en la sangre.

- Funcionamiento del Proceso de comprobación de Interacciones

La funcionalidad del proceso puede entenderse mejor a través de dos ejemplos hipotéticos. El primero corresponde al escenario preventivo y el segundo al escenario informativo.

- Escenario preventivo

Imaginemos al usuario llamado Juan que está tomando Enalapril, un inhibidor de la enzima convertidora de la angiotensina (ECA). Juan desea incluir frutas en su dieta, específicamente naranjas, plátanos y manzanas. Para asegurarse de que estos cambios dietéticos no interfieran con su medicamento, decide consultar al *Agente Conversacional*.

El proceso de comprobación de interacciones opera a través de tres niveles de verificación. En el primer nivel, el proceso busca posibles interacciones entre el alimento específico que Juan va a consumir y su medicamento. Por ejemplo, si Juan planea consumir 1 naranja, el algoritmo buscará interacciones directas entre la naranja y el Enalapril. Según Drier (2020), se identifica que el alimento que Juan desea incluir en su dieta puede interactuar con los inhibidores de la ECA, como el Enalapril. En este contexto, una vez que el algoritmo ha detectado una interacción entre el alimento y el medicamento, entra en acción el algoritmo de

recomendación. Esta función desempeña un papel crítico, ya que busca una alternativa viable y segura para el usuario, asegurándose de que el nuevo alimento sugerido no interfiera con el medicamento que está tomando. Esto se logra mediante el uso de un algoritmo de filtrado basado en contenido, el cual analiza y selecciona una opción alimenticia adecuada, teniendo en cuenta la ausencia de interacciones negativas con el fármaco.

Es importante destacar que, en el contexto de este trabajo, las recomendaciones se generan exclusivamente cuando se identifica una interacción entre fármaco y alimento, actuando así de forma preventiva para mitigar posibles riesgos para la salud del usuario. La estructura y el funcionamiento detallado de este algoritmo de recomendación se explorará a fondo en el Capítulo 4.

Si no se identifica ninguna interacción en el primer nivel de análisis, el proceso avanza al segundo nivel. En este nivel, el proceso de comprobación examina las interacciones potenciales entre el medicamento y la categoría general de alimentos a la que pertenecen los alimentos. Tomando un ejemplo hipotético, si Juan planea consumir una manzana, si la manzana no tiene interacción directa con los medicamentos, entonces busca el concepto padre que es 'frutas', el proceso de comprobación buscaría cualquier interacción potencial entre el medicamento que está tomando Juan y la categoría de 'frutas'. Este proceso implica que el algoritmo está programado para buscar interacciones más allá del nivel de alimentos individuales, extendiendo su análisis a grupos de alimentos más amplios para garantizar una evaluación de las posibles interacciones entre el medicamento y la dieta de Juan.

En un escenario alternativo, cuando la interacción no es directamente con el alimento específico ni con su grupo, el algoritmo se desplaza al tercer nivel de verificación. Este nivel se enfoca en analizar la interacción entre los componentes nutricionales del alimento a consumir y el fármaco. Por ejemplo, consideremos el caso de Juan tomando Enalapril y decidiendo consumir un plátano. El plátano no tiene una interacción directa con el Enalapril, sino su componente potasio. Por lo tanto, el proceso de comprobación, en el primer nivel, buscará interacciones directas entre el plátano y el Enalapril. No encontrando ninguna interacción en este nivel, el algoritmo se desplaza al segundo nivel de verificación, donde verifica a qué grupo pertenece el plátano, en este caso, las frutas, con este grupo verifica la interacción con el Enalapril. Nuevamente, si no se identifica ninguna interacción, el algoritmo avanza al tercer nivel.

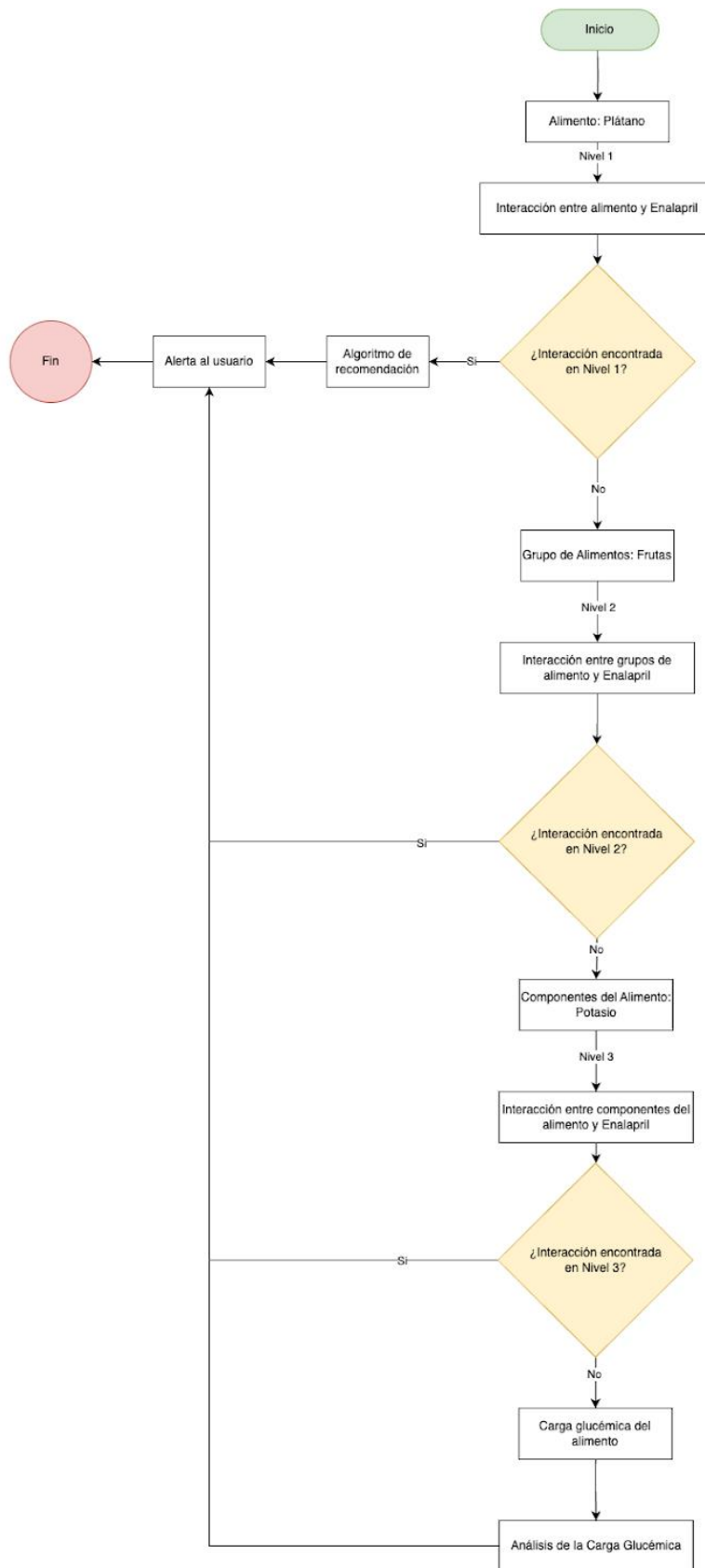
En este tercer nivel, el proceso de comprobación analiza las interacciones entre los componentes nutricionales del plátano y el Enalapril. Identificando potenciales interacciones entre el potasio, un componente nutricional del plátano, y el Enalapril. Al detectar esta

interacción, el algoritmo proporciona una descripción detallada de la misma y del medicamento involucrado. Posteriormente, se alerta a Juan sobre la posible interacción entre el Enalapril y el plátano, basándose en su contenido de potasio. De esta manera, Juan puede conocer sobre las interacciones del alimento con el fármaco, de esta forma puede tener precaución de consumir dicho alimento.

Continuando con el análisis, supongamos que ahora Juan tiene la intención de consumir tres manzanas. En este escenario, el proceso de comprobación realizará de nuevo la verificación en los tres niveles para determinar posibles interacciones entre las manzanas y el Enalapril. Tras una revisión, el proceso de comprobación no identificará ninguna interacción en ninguno de los tres niveles. Esto significa que no existen interacciones directas entre las manzanas y el Enalapril, ni con el grupo al que pertenecen las manzanas (frutas), ni tampoco con los componentes nutricionales presentes en las manzanas. Por lo tanto, hasta este punto, la interacción fármaco-alimento se considera inexistente y la búsqueda finaliza aquí.

No obstante, el análisis no concluye. Ahora, el proceso de comprobación recurre a la evaluación de la carga glucémica de las tres manzanas que Juan planea consumir. Para ello, se calcula la carga glucémica utilizando la fórmula previamente explicada. En este caso, la carga glucémica resultante para las tres manzanas es de 12.96. Según los rangos definidos anteriormente, esta carga glucémica se encuentra en el rango medio (10 a 19). Por ende, el proceso de comprobación generará un mensaje de precaución para el usuario, que será: 'Procura consumir con cuidado tres manzanas. Alto contenido en fósforo y fibra soluble'. De esta forma, el sistema guía a Juan para que pueda tomar decisiones más informadas acerca de su dieta, considerando tanto las posibles interacciones de su medicación con los alimentos, como la carga glucémica de estos, como se muestra en la Figura 19.

Figura 19 Niveles de identificación de interacción fármaco-alimento.



- Escenario informativo

En un segundo escenario, tenemos a Juan que padece un dolor de cabeza intenso y ha optado por tomar ibuprofeno, un medicamento de venta libre común para aliviar esta sintomatología; Juan no se siente bien y decide consultar al agente. El proceso de comprobación recupera todos los medicamentos que Juan ha estado tomando y le pregunta si ha consumido cualquier medicamento adicional. Juan menciona Ibuprofeno. El proceso de comprobación, entonces, consulta la base de datos de interacciones y examina la relación entre los medicamentos de Juan y el ibuprofeno. En este caso, descubre una interacción Glipizida-Ibuprofeno, por lo que se emite una alerta con una descripción detallada de dicha interacción.

3.3.2. Algoritmo de recomendaciones

Este componente proporciona las recomendaciones basadas en las interacciones del usuario y la información almacenada en la base de datos. Las recomendaciones están enfocadas a dar alternativas de alimentos, que son comunicados al agente conversacional para ser transmitidas al usuario. Una explicación más detallada sobre el diseño y funcionamiento del algoritmo de recomendación, se describe en el Capítulo 4.

3.4. API

El componente denominado API cumple con la función de suministrar los datos pertinentes para adquirir conocimiento sobre las interacciones farmacológicas. La recolección de dicha información se efectuó mediante técnicas de web scraping. En relación a la inclusión de los datos en las bases de datos, el administrador cuenta con un endpoint específico, a través del cual es posible incorporar la información recabada a partir del web scraping.

3.4.1. Web Scraping

El proceso de recopilación de información se realiza complementariamente con la técnica de web scraping aplicada a DrugBank. Esta fuente de datos ha sido seleccionada debido a su vasta información en cuanto a medicamentos (Wishart et al., 2018). En una evaluación comparativa, Vademecum⁸ también fue examinado, sin embargo, a pesar de proporcionar información sobre interacciones farmacológicas, esta se limitaba a interacciones fármaco-fármaco y carecía de detalles sobre fármaco-alimentos. Por este motivo, DrugBank se eligió como opción preferente.

Para almacenar los términos recuperados desde DrugBank usando web scraping, se desarrolló un script, del cual se ofrece una visión general simplificada en la Figura 20.

⁸<https://www.vademecum.es/>

Este código, al que se puede acceder a través del repositorio, permite extraer todas las interacciones asociadas a un medicamento específico. La información recopilada se almacena en un archivo de formato CSV.

Como ejemplo, al consultar el medicamento Abatacept se obtienen 848 interacciones con otros medicamentos, en la Figura 21 se muestra la estructura JSON que se obtiene al extraer los medicamentos al realizar web scraping. Para verificar esta extracción completa se puede verificar el archivo 'interactions.csv' en el repositorio⁹. Este archivo contiene la información del web scraping y así se formará para cualquier otro medicamento.

Para un análisis detallado de los parámetros utilizados, se recomienda consultar la documentación de la API, disponible en el repositorio¹⁰. Esta documentación también incluye detalles sobre un endpoint adicional dedicado a la adición de medicamentos de forma individual.

Figura 20 Script de implementación de web scraping.

```
async function getDrugId(drugTrans) {...}
async function fetchInteractions(drugId, start, length) {...}
async function fetchAllInteractions(drugTrans) {...}
async function saveInteractionsToJSON(drugName, interactions, fileName) {...}
async function processDrugInteractions(drugName, fileName) {
  return new Promise(async (resolve, reject) => {
    try {
      const interactions = await fetchAllInteractions(drugName);
      saveInteractionsToJSON(drugName, interactions, fileName);
      resolve({ success: true, message: 'Interacciones guardada correctamente.' });
    } catch (error) {
      reject({ success: false, message: 'Error al procesar las interacciones.',
        error });
    }
  });
}
processDrugInteractions("Abatacept", "interactions.json");
```

⁹<https://github.com/kambdan/WebscrappingDrug.git>

¹⁰<https://github.com/kambdan/APIdocumentation.git>

Figura 21 Ejemplo de la ejecución de web scrapping.

```
{
  "drug": "Abatacept",
  "interactions": [
    {
      "drug": "Abemaciclib",
      "interaction": "The metabolism of Abemaciclib can be increased when combined with
        Abatacept."
    },
    {
      "drug": "Abrocitinib",
      "interaction": "The metabolism of Abrocitinib can be increased when combined with
        Abatacept."
    },
    {
      "drug": "Acalabrutinib",
      "interaction": "The metabolism of Acalabrutinib can be increased when combined with
        Abatacept."
    }
  ]
}
```

3.4.2. Agregar Interacciones farmacológicas

Para este caso se ha desarrollado un endpoint que permite alimentar la base de datos. La Figura 22 muestra el endpoint que permite realizar una consulta POST para agregar una nueva interacción. La Figura 23 muestra la estructura del archivo JSON que debe enviarse. Para poder agregar las interacciones de los medicamentos.

Figura 22 Endpoint para agregar medicamentos con sus interacciones.

```
POST /api/v1/data/addDrugWithInteractions
```

Figura 23 Estructura JSON para el envío de datos.

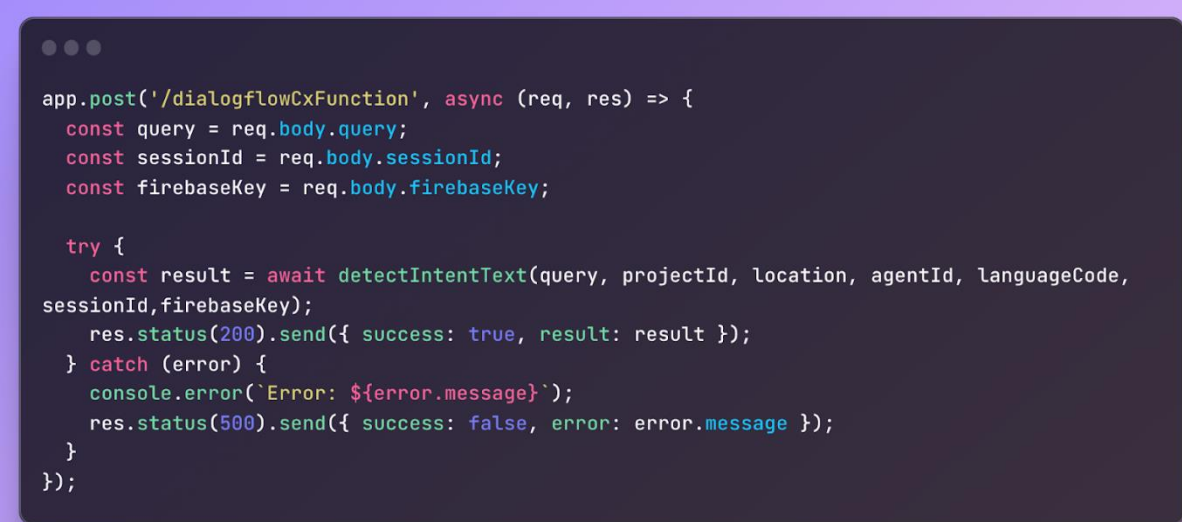
```
{
  "drug": "Abatacept",
  "interactions": [
    {
      "drug": "Abemaciclib",
      "interaction": "The metabolism of Abemaciclib can be increased when combined with
        Abatacept."
    }
  ]
}
```

3.5. API Interfaz de usuario

Este componente proporciona una interfaz unificada para que los usuarios interactúen con el agente conversacional. El agente conversacional utiliza esta API para transmitir las respuestas y recomendaciones a los usuarios. La Figura 24 proporciona un esquema simplificado de esta función. Es importante destacar que los parámetros del API incluyen:

- Query: Corresponde al mensaje enviado por el usuario al agente.
- Session id: Permite la diferenciación entre diversas sesiones y se gestiona en la aplicación móvil para cada usuario, permitiendo así la creación de nuevas sesiones de usuario.
- Firebase key: Se emplea para determinar a qué usuario corresponde cada sesión y en base a esta información, obtener sus respectivos medicamentos.

Figura 24 API para la interfaz de usuario.



```
app.post('/dialogflowCxFunction', async (req, res) => {
  const query = req.body.query;
  const sessionId = req.body.sessionId;
  const firebaseKey = req.body.firebaseKey;

  try {
    const result = await detectIntentText(query, projectId, location, agentId, languageCode,
    sessionId, firebaseKey);
    res.status(200).send({ success: true, result: result });
  } catch (error) {
    console.error(`Error: ${error.message}`);
    res.status(500).send({ success: false, error: error.message });
  }
});
```

3.6. Dispositivos externos

La expansión de la plataforma hacia dispositivos externos es un aspecto fundamental en la planificación y el diseño del proyecto. A través de la API de la Interfaz de Usuario, es posible vincular diferentes tipos de dispositivos y plataformas, extendiendo así la funcionalidad y accesibilidad de nuestra aplicación.

El proyecto, se ha focalizado en el desarrollo y perfeccionamiento de una aplicación móvil. Este objetivo ha sido logrado con éxito y la aplicación se encuentra plenamente funcional. Sin embargo, se ha procurado extender el funcionamiento inicial más allá del uso de dispositivos móviles.

Las primeras pruebas de interoperabilidad se han realizado tanto en plataformas web como con dispositivos de la gama Alexa. Sin embargo, es importante aclarar que estas plataformas aún no están listas para una implementación completa y su uso público. Esta realidad se debe, en gran parte, a desafíos relacionados con la autenticación de los usuarios.

La autenticación es un aspecto clave de la seguridad y la privacidad del usuario. En el caso de nuestra aplicación móvil, hemos optado por utilizar Firebase para la autenticación del usuario. Sin embargo, al abordar la implementación en plataformas web y Alexa, se encontró que la ausencia de una interfaz para la autenticación a través de Firebase representa un obstáculo significativo.

Esta falta de una interfaz unificada para la autenticación presenta un desafío en términos de consistencia y seguridad. Debido a este desafío, estas plataformas aún no están preparadas para una implementación completa y requieren un análisis y desarrollo adicionales para garantizar una autenticación segura y efectiva.

En futuros trabajos, se puede abordar este desafío y desarrollar una solución para la autenticación en estas plataformas. Este avance permitirá una mayor interoperabilidad y abrirá nuevas posibilidades para la aplicación, ampliando su alcance más allá de los dispositivos móviles y permitiendo un acceso más amplio para los usuarios.

3.6.1. Aplicación Móvil

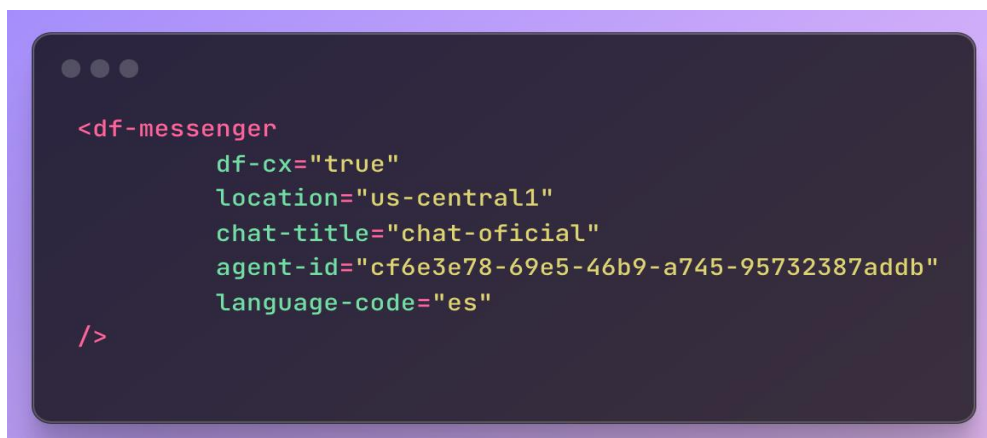
La construcción del aplicativo móvil es una parte integral del proyecto, que integra un agente conversacional implementado en Flutter. El objetivo es mejorar la interacción entre usuarios y el agente, brindando una experiencia móvil única y accesible. La interfaz del aplicativo permite un acceso rápido, sencillo al agente, y con capacidad de interacción mediante texto y voz. Información detallada sobre el diseño e implementación del aplicativo móvil puede ser consultado en el Capítulo 5.

3.6.2. Web

La Figura 25 ilustra una característica significativa proporcionada por Dialogflow CX, que es su capacidad para integrarse con interfaces web. En el caso específico de este proyecto, el sitio web propuesto invoca un plugin preconfigurado con las credenciales pertinentes para permitir dicha integración. Esta funcionalidad ha sido probada, sin embargo, como el agente está enfocado a ser personalizado en cuanto a los medicamentos que lleva tomando no se puede interactuar de esta manera. Para lograr un nivel de personalización, se recomienda la implementación de un mecanismo de autenticación que pueda interactuar con Firebase. Este

mecanismo permitiría el almacenamiento de información personalizada del usuario, como sus alimentos consumidos, mejorando así su experiencia al usar la herramienta.

Figura 25 Implementación en la Web.

A screenshot of a code editor with a dark background and light-colored text. The code is XML, defining a messenger widget. The code is as follows:

```
<df-messenger
  df-cx="true"
  location="us-central1"
  chat-title="chat-oficial"
  agent-id="cf6e3e78-69e5-46b9-a745-95732387addb"
  language-code="es"
/>
```

3.6.3. Alexa

Otra fase de prueba que se ha llevado a cabo durante este proyecto implica el desarrollo de código¹¹, con el propósito de conectar Alexa con la *API Interfaz de Usuario*. Esta acción representa un esfuerzo inicial para explorar las posibilidades de interoperabilidad entre la aplicación propuesta y los asistentes de voz, como Alexa, que se han vuelto cada vez más ubicuos en la vida cotidiana.

A pesar de la exitosa implementación de esta conexión inicial, se reconoce que el logro de una integración completa y personalizada con Alexa requerirá etapas de desarrollo adicionales. Una de estas etapas críticas implica la implementación de un mecanismo de autenticación. Al igual que con la integración web discutida anteriormente, este mecanismo facilitaría el almacenamiento y recuperación de información personalizada del usuario, proporcionando así una experiencia de usuario más individualizada y enriquecida. En la Figura 26 se muestra parte del script.

¹¹<https://github.com/kambdan/WebscrappingDrug.git>

Figura 26 Script de conexión entre el agente y Alexa.

```
const Alexa = require('ask-sdk-core');
const axios = require('axios');
const LaunchRequestHandler = {...};
const HelpIntentHandler = {...};
const CancelAndStopIntentHandler = {...};

const FallbackIntentHandler = {...};

const bot = {
  canHandle(handlerInput) {
    return Alexa.getRequestType(handlerInput.requestEnvelope) === 'IntentRequest'
      && Alexa.getIntentName(handlerInput.requestEnvelope) === 'Bot';
    //return true;
  },
  async handle(handlerInput) {...};
const SessionEndedRequestHandler = {...};
const ErrorHandler = {...};

async function fetchResponseFromApi(userInput) {
  try {
    const apiUrl = 'https://deploy/dialogflowCxFunction/';
    const response = await axios.post(apiUrl, { query: userInput });
    const responseMessages = response.data.result.responseMessages;
    const flattenedMessages = responseMessages.flat();
    const messageString = flattenedMessages.join(' ');
    return messageString;
  } catch (error) {
    console.error(`Error al obtener respuesta de la API: ${error}`);
    return null;
  }
}

exports.handler = Alexa.SkillBuilders.custom()
  .addRequestHandlers(
    LaunchRequestHandler,
    HelpIntentHandler,
    CancelAndStopIntentHandler,
    FallbackIntentHandler,
    bot,
    SessionEndedRequestHandler)
  .addErrorHandlers(
    ErrorHandler)
  .withCustomUserAgent('sample/hello-world/v1.2')
  .lambda();
```


Capítulo 4 – DISEÑO DEL ALGORITMO DE RECOMENDACIÓN

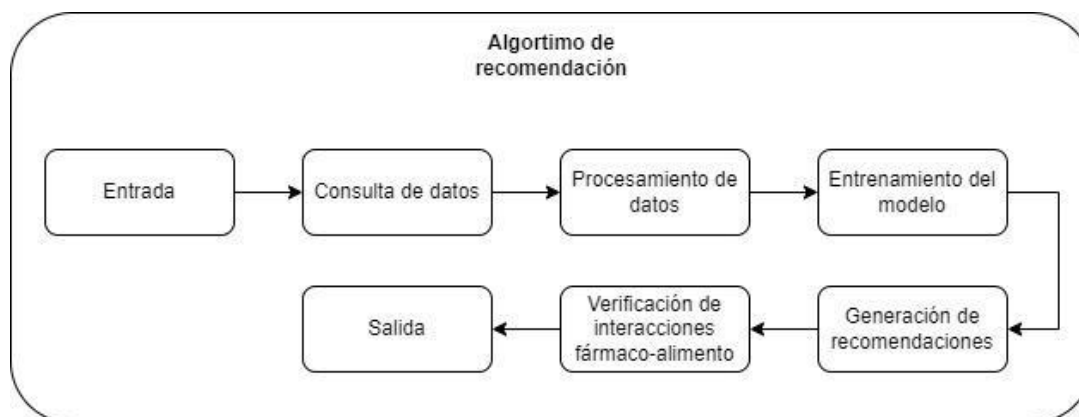
Este capítulo se centra en el diseño del algoritmo de recomendación de alimentos. El algoritmo es una herramienta sumamente importante que tiene como objetivo proporcionar recomendaciones de alimentos a los usuarios, teniendo en cuenta su medicación y evitando interacciones fármaco-alimento potencialmente peligrosas.

También, se describe en detalle cómo se desarrolló el algoritmo, explicando las decisiones tomadas en cada paso del proceso. Esto incluirá una discusión sobre cómo se seleccionaron y prepararon los datos de entrada, cómo se entrenó el modelo de aprendizaje automático y cómo se usa para generar recomendaciones. Adicionalmente, se detalla cómo se verifica la seguridad de las recomendaciones con respecto a las interacciones con medicamentos.

El algoritmo de filtrado, basado en contenido, es una técnica de recomendación que utiliza las características de los elementos para ofrecer sugerencias personalizadas a los usuarios, basándose en la premisa de que estos tienden a poseer preferencias consistentes y disfrutan de elementos con características similares a aquellos que han disfrutado previamente (Sánchez et al., 2017). La implementación de esta técnica permite al usuario conocer alternativas de consumo de alimentos, a partir de sus propiedades nutricionales.

Para implementar el filtrado basado en el contenido, se ha elegido utilizar Python debido a su robustez en el tratamiento de datos, aprendizaje automático e inteligencia artificial. Python también cuenta con una amplia gama de bibliotecas que facilitan el desarrollo de algoritmos complejos (Hao & Ho, 2019). A partir del uso de este lenguaje de programación, se definieron las fases de la implementación y funcionamiento del algoritmo, ilustrados en la Figura 27.

Figura 27 Fases de implementación y funcionamiento del algoritmo de recomendación.



4.1. Entrada

El algoritmo recibe un objeto JSON que contiene dos campos: 'alimento' y 'medicamentos' (Figura 28).

- Alimento: Es el nombre del alimento del que se buscan alternativas de consumo.
- Medicamentos: Representa la lista de medicamentos que el usuario está tomando actualmente.

Figura 28 Ejemplo de entrada del algoritmo de recomendación.

```
{
  "alimento": "Plátano",
  "medicamentos": ["Insulina", "Enalapril"]
}
```

4.2. Consulta de datos

El primer paso en la consulta de datos es obtener el grupo de alimentos al que pertenece el alimento proporcionado por el usuario. Esto se hace utilizando la función `getGrupoAlimento()` ilustrado en la Figura 29, que consulta la base de datos y devuelve el grupo de alimentos al que pertenece el alimento. Esto es importante porque los alimentos de un mismo grupo suelen tener propiedades nutricionales similares.

Figura 29 Código de implementación del método `getGrupoAlimento()`.

```
def getGrupoAlimento(alimento):
    try:
        response = supabase.table('Alimento').select('grupo_alimento')
            .eq('nombre', alimento).execute()
        if not response:
            return None
        return response['data'][0]['grupo_alimento']
    except Exception as e:
        print(f'Ocurrió un error: {e}')
        return None
```

Una vez que se tiene el grupo de alimentos, se utiliza la función `getAlimentos()`, que recupera todos los alimentos que pertenecen a ese mismo grupo de la base de datos (Figura 30). Los

datos recuperados para cada alimento incluyen su nombre, hidratos de carbono, índice glucémico y carga glucémica. Estos datos se almacenan en un diccionario de Python para su posterior uso.

Figura 30 Código de implementación del método `getAlimentos()`.

```
def getAlimentos(alimento):
    grupoAlimento = getGrupoAlimento(alimento)
    try:
        response = supabase.table('Alimento').
            select('id,nombre,hidratos_carbono,indice_glucemico,carga_glucemica')
            .eq('grupo_alimento', grupoAlimento).execute()
        if not response:
            return None
        return response['data']
    except Exception as e:
        print(f'Ocurrió un error: {e}')
        return None
```

La recuperación de datos es una etapa fundamental del algoritmo, ya que la información que se obtiene en este punto es lo que alimenta el resto del algoritmo. Es esencial que esta etapa se realice correctamente para garantizar la precisión de las recomendaciones generadas.

4.3. Preprocesamiento de datos

El preprocesamiento de datos es una fase crítica en cualquier algoritmo de aprendizaje automático, ya que prepara los datos brutos para ser introducidos en el modelo. Los detalles del preprocesamiento en este algoritmo son los siguientes:

Primero, los datos recuperados en el paso anterior se procesan en la función `getData()`, como muestra la Figura 31. Los datos de cada alimento se almacenan en un diccionario de Python, donde el nombre del alimento es la clave y los valores son una lista que contiene los hidratos de carbono, el índice glucémico y la carga glucémica de cada alimento. Un ejemplo de esto se observa en la Figura 32.

Figura 31 Código de implementación del método `getData()`.

```
def getData(alimento):  
    result = getAlimentos(alimento)  
    if result is None:  
        return None  
    else:  
        diccionarioAlimentos = {  
            fila['nombre']: [fila['hidratos_carbono'],  
                fila['indice_glucemico'], fila['carga_glucemica']]  
            for fila in result  
        }  
        return diccionarioAlimentos
```

Figura 32 Preprocesamiento de los datos (alimentos).

```
alimentos = {'Albaricoque': [9.5, 34, 3.2], 'Plátano': [58, 31, 17.9]}
```

Una vez que los datos se han organizado en este formato, se pueden preparar para su entrada en el modelo de aprendizaje automático, mediante la función `getRecommendation()`, con esto es necesario transformar los datos para que tengan una escala común.

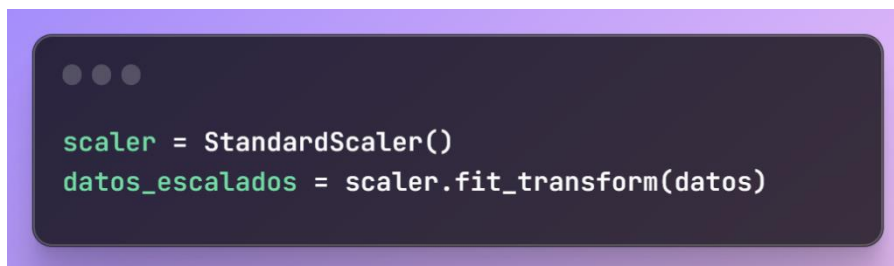
Los datos de diferentes características pueden tener distintos rangos. Por ejemplo, los hidratos de carbono pueden oscilar entre 0 y 100, mientras que el índice glucémico puede oscilar entre 0 y 1. Si se introduce directamente en el modelo de aprendizaje automático, estas diferencias en la escala pueden hacer que el modelo de demasiada importancia a ciertas características simplemente porque sus valores son más altos.

Para evitarlo, se utiliza una técnica llamada estandarización. La estandarización se emplea para transformar características en un conjunto de datos de manera que tengan una distribución con media cero y una desviación estándar de uno. Al restar la media de cada característica, se centra alrededor del valor cero, lo que ayuda a eliminar cualquier sesgo o tendencia en los datos. Luego, al dividir para la desviación estándar, se escala la

característica para que su variabilidad sea comparable a otras características. Esta técnica se usa para evitar que las diferencias en las magnitudes o escalas de las características afecten el proceso de aprendizaje automático.

El algoritmo implementa la clase `StandardScaler` de la biblioteca `Scikit-learn` para realizar dicho proceso, es decir, se ajusta el escalador a los datos de los alimentos, y luego se utiliza para transformarlos (Figura 33). De esta manera, los datos están preparados para ser introducidos en el modelo de aprendizaje automático.

Figura 33 Estandarización de los datos.

A screenshot of a code editor showing two lines of Python code. The first line is `scaler = StandardScaler()` and the second line is `datos_escalados = scaler.fit_transform(datos)`. The code is displayed in a dark-themed editor with a light purple border.

```
scaler = StandardScaler()
datos_escalados = scaler.fit_transform(datos)
```

4.4. Entrenamiento del modelo

En este algoritmo se entrena un modelo de red neuronal utilizando TensorFlow, se trata de una biblioteca de código abierto desarrollada por Google para realizar cálculos numéricos de manera eficiente y está especialmente diseñada para Machine Learning (TensorFlow, 2023). El modelo de red neuronal, también conocido como red neuronal artificial, es una estructura computacional inspirada en el funcionamiento del cerebro humano. Está compuesto por unidades interconectadas llamadas neuronas artificiales o nodos, organizadas en capas (García, 2021).

En un modelo de red neuronal, las neuronas están agrupadas en capas, y la información fluye a través de ellas desde la capa de entrada, a través de una o varias capas ocultas, hasta la capa de salida. Cada neurona realiza cálculos en base a las entradas recibidas y aplica una función de activación para producir una salida. Las conexiones entre las neuronas, llamadas pesos, determinan la fuerza y dirección de la influencia que una neurona tiene sobre otra (García, 2021). La red neuronal del algoritmo se define en la función `getRecommendation()`. Se utiliza la clase `Sequential` de TensorFlow para crear una red neuronal que consta de tres capas (Figura 34).

Figura 34 Modelo de la red neuronal del algoritmo de recomendación.

```
model = Sequential()  
model.add(Dense(64, activation='relu', input_shape=(datos_escalados.shape[1],)))  
model.add(Dense(32, activation='relu'))  
model.add(Dense(datos_escalados.shape[1], activation='linear'))
```

- La primera capa, o capa de entrada, tiene 64 neuronas y utiliza la función de activación ReLU (Rectified Linear Unit). La función ReLU es una función no lineal que da como salida el valor de entrada si es positivo, y cero en caso contrario. Esta función de activación ayuda a la red neuronal a aprender patrones complejos en los datos.
- La segunda capa tiene 32 neuronas y también utiliza la función de activación ReLU.
- La tercera capa, o capa de salida, tiene el mismo número de neuronas que el número de características en los datos de entrada (en este caso, tres: hidratos de carbono, índice glucémico y carga glucémica), y utiliza una función de activación lineal. La función de activación lineal da como salida el valor de entrada sin cambios, lo que permite que la red neuronal produzca cualquier rango de valores en la salida.

Una vez que se ha definido el modelo, se compila con la función de pérdida de error cuadrático medio y el optimizador Adam (Figura 35). La función de pérdida de Error Cuadrático Medio (ECM) sirve para evaluar la discrepancia entre los valores predichos por el modelo y los valores reales de los datos, mientras que el optimizador Adam ajusta los parámetros del modelo para minimizar esta diferencia (Cuevas et al., 2020).

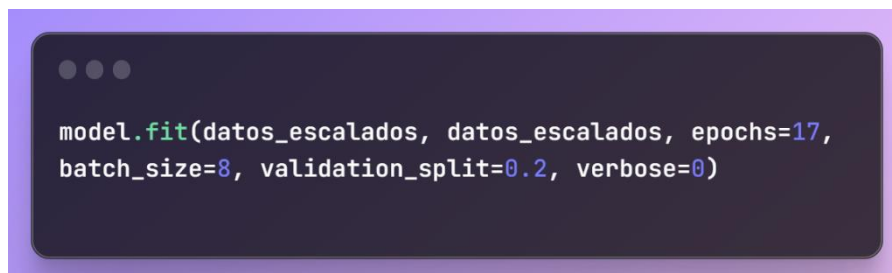
Figura 35 Compilación del modelo del algoritmo de recomendación.

```
model.compile(loss='mean_squared_error', optimizer='adam')
```

Finalmente, se entrena el modelo en los datos de los alimentos utilizando un conjunto de entrenamiento y un conjunto de validación. El conjunto de entrenamiento es un subconjunto de los datos que se utiliza para ajustar los parámetros del modelo, mientras que el conjunto de validación es un subconjunto separado que se utiliza para evaluar el rendimiento del modelo durante el entrenamiento (Chanampe et al., 2019). En este caso, se utiliza un 80%

de los datos para el entrenamiento y un 20% para la validación. El modelo se entrena durante 17 épocas, que es el número de veces que el modelo pasa por todo el conjunto de entrenamiento (Figura 36).

Figura 36 Entrenamiento del modelo del algoritmo de recomendación.

A screenshot of a code editor with a dark background and light text. The code is written in Python and shows the fit method of a model. The parameters are: datos_escalados, datos_escalados, epochs=17, batch_size=8, validation_split=0.2, and verbose=0.

```
model.fit(datos_escalados, datos_escalados, epochs=17,  
batch_size=8, validation_split=0.2, verbose=0)
```

En síntesis, el proceso de entrenamiento del modelo involucra la definición de la arquitectura de la red neuronal, la compilación del modelo con una función de pérdida y un optimizador, y el entrenamiento del modelo en los datos de los alimentos. El resultado es un modelo que ha aprendido a predecir los atributos de los alimentos a partir de sus hidratos de carbono, índice glucémico y carga glucémica.

4.5. Generación de recomendaciones

La fase de generación de recomendaciones se lleva a cabo después de la etapa de entrenamiento del modelo. En este contexto, una vez que la red neuronal ha sido meticulosamente entrenada, se despliega para predecir los atributos de un alimento específico que el usuario ha identificado. En este procedimiento, se podrían utilizar algoritmos de similitud, tales como: distancia euclidiana, distancia de Manhattan, distancia de Levenshtein, similitud del coseno, entre otros.

En el algoritmo se aplica la similitud del coseno, una métrica sofisticada, para determinar la cercanía entre el alimento especificado y las demás opciones de la base de datos, lo que ayuda a generar recomendaciones más acertadas y personalizadas para el usuario. Generalmente esta técnica se utiliza en la recuperación de información y algoritmos de recomendación (Olguin et al., 2019). En términos matemáticos, se define como el coseno del ángulo entre dos vectores.

Para el caso del algoritmo, se calcula la similitud del coseno entre los atributos predichos del alimento proporcionado por el usuario y los atributos de todos los alimentos en la base de datos. Cada alimento se representa como un vector tridimensional que contiene sus atributos de hidratos de carbono, índice glucémico y carga glucémica. Así, la similitud de coseno mide la similitud nutricional entre los alimentos.

Figura 37 Cálculo de similitudes de alimentos mediante la similitud del coseno.

```
usuario_escalado = scaler.transform([usuario])
atributos_predichos_escalados = model.predict(usuario_escalado)
atributos_predichos = scaler.inverse_transform(atributos_predichos_escalados)
similitudes = cosine_similarity(atributos_predichos, datos)
```

La Figura 37 muestra el código implementado de cómo se escalan los atributos del alimento del usuario para que estén en la misma escala que los datos en los que se entrenó el modelo. Luego, el modelo de la red neuronal se utiliza para predecir los atributos del alimento del usuario; las predicciones se ajustan a la escala de los datos originales para que estén en la misma unidad de medida. Finalmente, se calcula la similitud del coseno entre los atributos predichos y los atributos de todos los alimentos en la base de datos.

Esto proporciona una lista de valores de similitud del coseno en una escala [0-1], cada uno correspondiente a un alimento en la base de datos. Los valores más cercanos a 1 indican una mayor similitud con el alimento del usuario, mientras que los valores más bajos indican una menor similitud como ilustra la figura antes mencionada.

Figura 38 Lista de valores obtenidos mediante la similitud del coseno.

```
[ 0.99646063 0.72222032 0.64628651 0.99195952 0.9986087 0.94280047
0.95959889 0.98063732 0.99855743 0.99806005 0.83474685 0.99807634
0.99941666 0.79147807 0.86599097 0.83025438 0.99458917 0.99265996
0.99677116 0.99658339 0.99165002 0.99212552 0.99195952 0.99450295
0.99938703 0.99649191 0.99742085 0.99824409 0.97733937 0.69468399
0.99201864 0.99926272 0.93754523 0.83431616 0.97227 0.98080143
0.9866233 0.99348189 0.99076287 0.99703139 0.99008747 0.99425256
0.99703139 0.85500584 0.84408329 0.99755979 0.99810546 0.99982685
0.99520307 0.96565976 0.99915604 0.99452134 0.98656764 ]
```

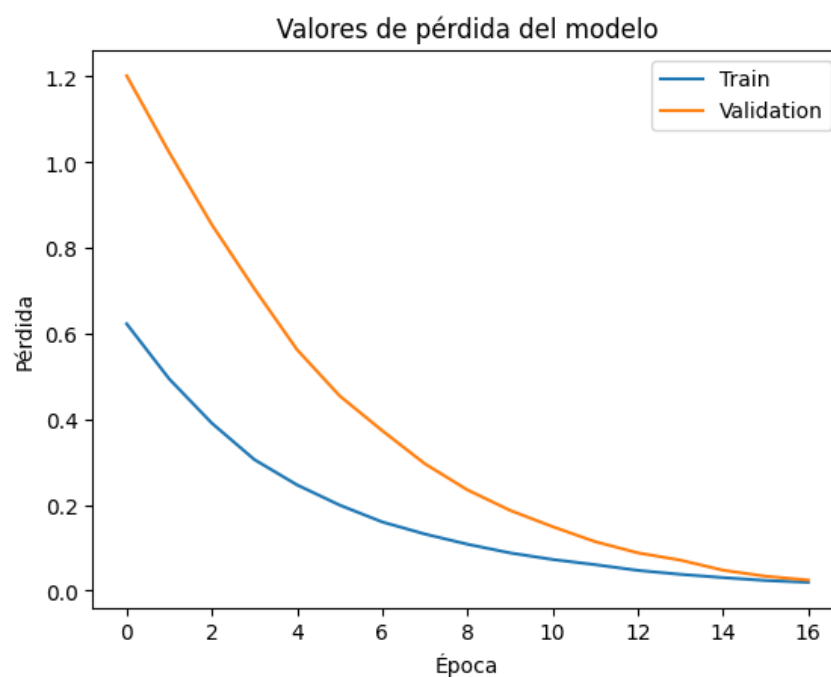
Después de calcular la similitud, los alimentos se ordenan en función de ella con el alimento del usuario, de mayor a menor. Para lograrlo, se utilizan los índices de los alimentos para ordenar la lista de sus nombres, y finalmente, se seleccionan aquellos más similares al alimento dado por el usuario como recomendaciones. En este caso, se excluye el alimento del usuario de las recomendaciones porque no tiene sentido recomendar el mismo alimento que el usuario ya está comiendo (Figura 39).

Figura 39 Selección y recomendación de alimentos similares al plátano.

```
Alternativas de consumo para Plátano :  
['Coco fresco', 'Litchi', 'Melocotón conserva', 'Uva', 'Caqui', 'Chirimoya', 'Manzana']
```

Para verificar el algoritmo de recomendación, se determinaron los valores de pérdida, tanto para los datos de entrenamiento como los de validación, obteniendo 0.028 y 0.0047 respectivamente (Figura 40). Un valor de pérdida de entrenamiento de 0.0028 significa que, en promedio, las predicciones del modelo para el conjunto de entrenamiento se desvían del valor real por aproximadamente 0.0028. Esto es bajo, lo que indica que el modelo está haciendo un buen trabajo al aprender de los datos de entrenamiento.

Figura 40 Evaluación del algoritmo de recomendación.



Un valor de pérdida de validación de 0.0047, por otro lado, significa que las predicciones del modelo para el conjunto de validación se desvían del valor real por aproximadamente 0.0047. Este valor es más alto que el valor de pérdida de entrenamiento, sin embargo, dado que el modelo no se entrena en los datos de validación, es normal que su rendimiento sea menor en comparación a los del entrenamiento.

No obstante, la pérdida de validación sigue siendo baja, lo que indica que el modelo está haciendo un buen trabajo, al generalizar a datos nuevos que no se utilizaron durante el entrenamiento. Si la pérdida de validación fuera significativamente más alta que la pérdida de entrenamiento, esto podría indicar que el modelo está sobre ajustado a los datos de entrenamiento y no está generalizando bien a los datos nuevos, pero este no es el caso. Los valores de pérdida indican que el modelo es eficiente, tanto en el aprendizaje de los datos de entrenamiento, como en la generalización de datos nuevos.

4.6. Verificación de interacciones fármaco-alimento

Después de generar las recomendaciones preliminares de alimentos alternativos, el algoritmo verifica si cada alimento recomendado tiene alguna interacción con los medicamentos que el usuario está tomando. Este proceso se lleva a cabo en la función `getApilInteractions()`, cuyos parámetros son: el alimento del usuario y su lista de medicamentos, tal como muestra la Figura 28. A continuación, se describen los siguientes pasos:

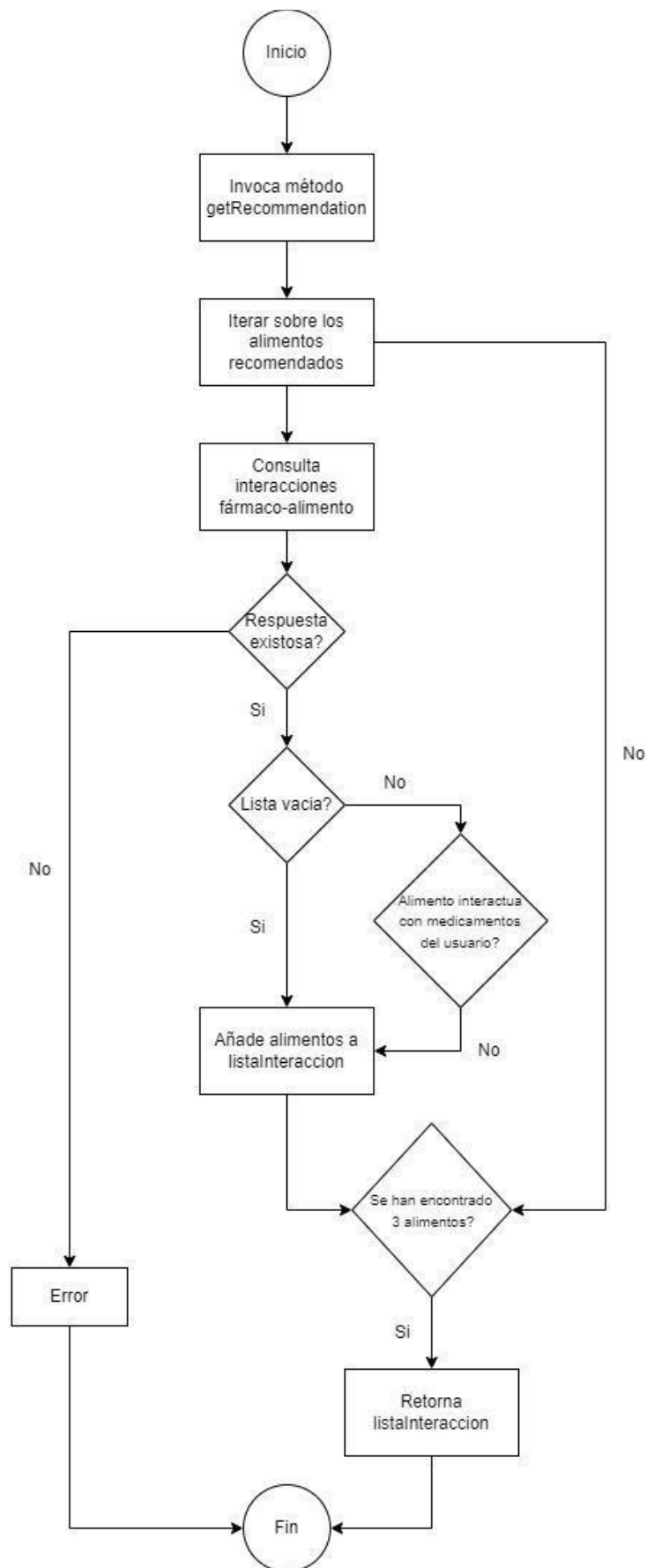
- Generación de una lista preliminar de alimentos recomendados: La función `getRecommendation()` es invocada para obtener una lista preliminar de alimentos que comparten características nutricionales con el alimento proporcionado por el usuario. Estas características incluyen hidratos de carbono, índice glucémico y carga glucémica.
- Iteración a través de la lista de recomendaciones: La función procede a iterar sobre cada alimento en la lista de recomendaciones, con el objetivo de verificar si existe alguna interacción conocida con los medicamentos del usuario.
- Consulta de interacciones alimento-medicamento: Para cada alimento en la lista, la función realiza una consulta a una base de datos externa. Esta consulta busca obtener información acerca de cualquier posible interacción entre el alimento en cuestión y los medicamentos proporcionados por el usuario.
- Análisis de las respuestas de la consulta: La respuesta de cada consulta es analizada cuidadosamente. En caso de una respuesta exitosa (indicada por un código de estado 200), la respuesta es convertida a formato JSON y posteriormente se analiza para obtener información relevante acerca de las

interacciones farmacológicas. Si la respuesta contiene una lista vacía, se interpreta como que el alimento no presenta interacciones conocidas con ningún medicamento. En ese caso, el alimento es añadido a la lista 'listaInteraccion', que a priori será la salida del algoritmo. En caso de que la respuesta contenga información sobre interacciones, se realiza una comprobación adicional para determinar si el alimento interactúa con alguno de los medicamentos proporcionados por el usuario. Si no hay interacción, el alimento también es añadido a la lista 'listaInteraccion'.

- Criterio de terminación anticipada: El proceso de iteración y consulta se detiene una vez que se han identificado tres alimentos que no presentan interacciones con los medicamentos del usuario. Esto se hace con el fin de optimizar la eficiencia del algoritmo y proporcionar una respuesta rápida al usuario.
- Devolución de la lista de alimentos recomendados: Al finalizar el proceso, la función devuelve la lista 'listaInteraccion'. Esta lista contiene hasta tres alimentos que, según la información disponible, son similares al alimento original proporcionado por el usuario y no presentan interacciones con los medicamentos que el usuario está tomando.

También, se presenta el diagrama de flujo que ilustra el proceso requerido para verificar las interacciones entre fármacos y alimentos.

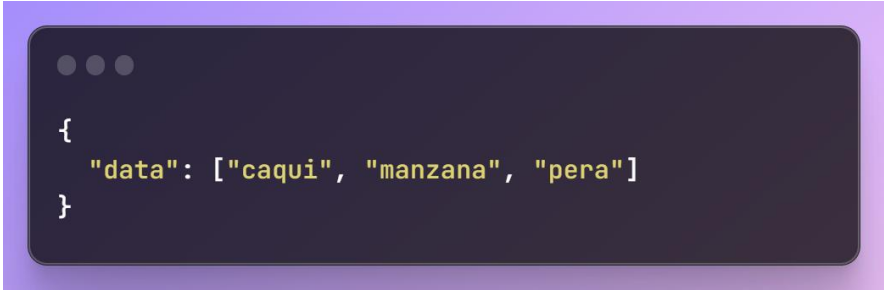
Figura 41 Flujo de verificación de interacciones fármaco-alimento.



4.7. Salida

La salida final del algoritmo es la respuesta de la función `getApiInteractions()`. Esta respuesta es una lista de hasta tres alimentos que son nutricionalmente similares al alimento original proporcionado por el usuario y que no interactúan con los medicamentos que esté tomando (Figura 42).

Figura 42 Salida del algoritmo de recomendación.



```
{
  "data": ["caqui", "manzana", "pera"]
}
```

Se puede observar que la respuesta es un objeto JSON con un solo campo, `data`, que contiene una lista de alimentos. Cada elemento de la lista es una cadena de texto que representa el nombre de un alimento. Esta respuesta se genera de la siguiente manera en la Figura 43.

Figura 43 Código para generar la salida del algoritmo.



```
alimentos = getApiInteractions(alimento, medicamentos)
if alimentos is None:
    return f'No data para "{alimento}"'
return {'data': alimentos}
```

Si la función `getApiInteractions()` devuelve `'None'`, lo que significa que no se encontraron alimentos que cumplieran los criterios, se devuelve un mensaje de error indicando que no se encontraron datos para el alimento proporcionado por el usuario. En cambio, si la función devuelve una lista de alimentos, se devuelve el objeto JSON.

Esta respuesta puede ser utilizada por una aplicación o servicio de cliente para proporcionar recomendaciones de alimentos a los usuarios con base a sus necesidades dietéticas y a los medicamentos que están tomando. Por ejemplo, si un usuario está buscando alimentos que sean similares a un plátano, pero que no interactúen con la Insulina y el Enalapril, puede

recibir como respuesta la lista de la Figura 42, lo que significa que estos tres alimentos son buenos candidatos para reemplazar al plátano en su dieta.

A partir del diseño del agente conversaciones y el algoritmo de recomendación, se incorporan dentro de una aplicación móvil, que será detallado en el próximo capítulo.

Capítulo 5 – APLICACIÓN MÓVIL

El presente capítulo introduce el aplicativo móvil desarrollado como parte integral de este trabajo, el cual integra el agente conversacional implementado mediante la tecnología Flutter. Esta solución innovadora ha sido diseñada con el objetivo de potenciar la interacción entre los usuarios y el agente conversacional, brindando una experiencia móvil única y accesible.

En la actualidad, los aplicativos móviles han revolucionado la forma de relacionarse con la tecnología, ofreciendo una amplia gama de posibilidades para mejorar nuestra vida cotidiana. Siguiendo esta tendencia, se ha decidido implementar el agente conversacional dentro de un aplicativo móvil con el propósito de proporcionar a los usuarios una experiencia más cómoda y personalizada. La interfaz del aplicativo móvil desarrollado se caracteriza por su amigabilidad y atractivo visual, permitiendo a los usuarios acceder de manera rápida y sencilla al agente conversacional. Mediante la adecuada selección de elementos de diseño y funcionalidades, se ha buscado garantizar una experiencia de usuario fluida y agradable.

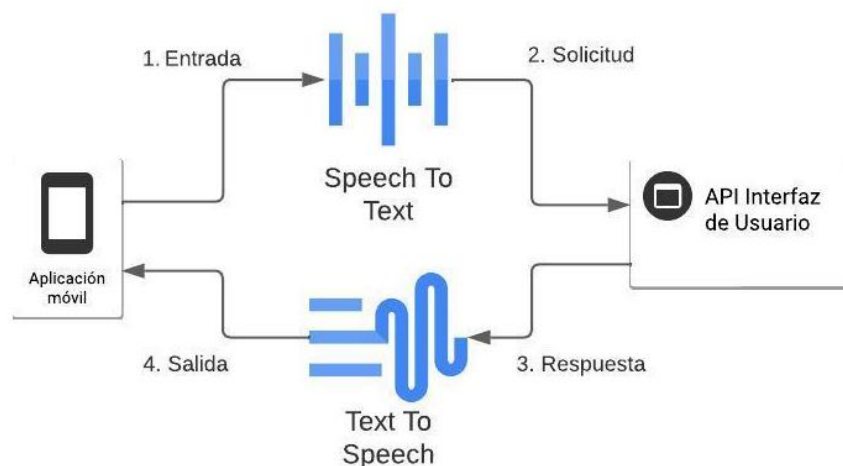
Es importante destacar que el aplicativo móvil es compatible con los sistemas operativos iOS y Android, lo que garantiza una amplia cobertura y accesibilidad para los usuarios, sin importar el dispositivo móvil que utilicen. En cuanto a la construcción del prototipo de la aplicación, se optó por utilizar Flutter, un framework que ofrece facilidades para desarrollar aplicaciones multiplataforma. Dentro de este framework, se ha implementado la capacidad de interacción mediante texto y voz.

Para la interacción por voz, se han seleccionado dos bibliotecas específicas con el objetivo de mejorar la experiencia del usuario. La primera biblioteca utilizada, Speech To Text (versión 6.1.1) permite la conversión de voz a texto, este complemento consta de un conjunto de clases que agilizan la implementación de las capacidades de reconocimiento de voz nativas de la plataforma en Flutter. Su diseño se orienta principalmente hacia el reconocimiento de comandos y frases cortas, no a la transcripción continua del habla o a la funcionalidad de escucha constante (Speech to text, 2019).

La segunda biblioteca, Text To Speech (versión 3.6.3) permite la conversión de texto a voz, permitiendo al agente conversacional comunicarse de manera oral con los usuarios (Flutter tts, 2018). La integración de estas bibliotecas en el aplicativo móvil asegura una interacción más natural y versátil, brindando opciones tanto para la entrada de información por parte del usuario como para la respuesta del agente conversacional en formato de voz. Esto contribuye a una experiencia de usuario más intuitiva y enriquecedora.

La Figura 44 representa el flujo para obtener la información a través de las bibliotecas mencionadas.

Figura 44 Flujo de intercambio de información.



Nota: Arquitectura del flujo del intercambio de información, adaptado de (Google Cloud, s.f.)

- Si el usuario opta por ingresar información mediante la voz, tiene la opción de activar uno de los dos escenarios disponibles: preventivo o informativo. Utilizando la biblioteca Speech to Text, la voz capturada es convertida en texto. Posteriormente, este texto es enviado a la API Interfaz de Usuario (sección 3.5) para dar continuidad a la conversación. La aplicación ejecutará los procedimientos correspondientes y obtendrá las instrucciones solicitadas.
- Una vez que el agente ha completado todo el flujo para obtener las solicitudes del usuario, el resultado se emite en forma de texto a los servicios de la API Interfaz de Usuario (sección 3.5). A continuación, en la aplicación, este texto se transforma en audio, lo que permite al usuario recibir la respuesta a su solicitud de manera auditiva.

5.1. Diseño de la interfaz de usuario

La interfaz de usuario representa la parte gráfica de la aplicación móvil desarrollada y permite realizar consultas, ver recomendaciones de manera preventiva o informativa, basada en la interacción fármaco-alimento y fármaco-fármaco, esto de acorde a las necesidades específicas. Se describen las siguientes interfaces:

- Bienvenida

Esta parte incluye la bienvenida para el usuario y una descripción general de la aplicación móvil. La cual se muestra en la Figura 45.

Figura 45 Interfaz de usuario de bienvenida.



- Autenticación

El proceso de autenticación de usuarios en la aplicación móvil como se ve en la Figura 46, es posible gracias a la implementación de una interfaz ofrece iniciar sesión a través de los servicios de Google, permitiendo al usuario ingresar sus credenciales de autenticación, tales como su correo y contraseña. Una vez ingresado sus datos, la interfaz de usuario establece comunicación con Firebase para realizar la verificación y autenticación del mismo. Cabe destacar que, en caso de que la autenticación sea exitosa, el usuario tendrá acceso a todas las funcionalidades de la aplicación. Este proceso es fundamental para garantizar su seguridad y privacidad.

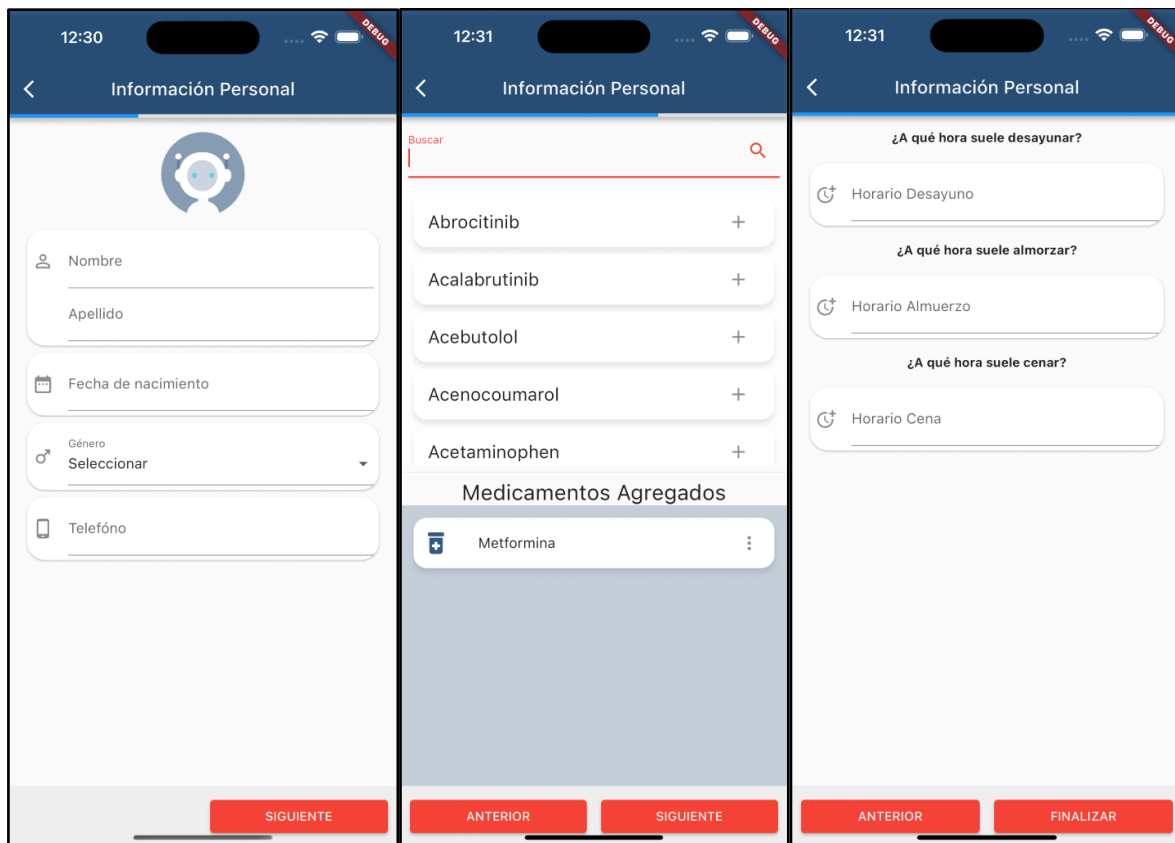
Figura 46 Autenticación del usuario.



- Registro

En la figura 47 se detalla el proceso de registro, como parte importante se toma los medicamentos que lleva consumiendo el usuario y los horarios de comidas. Los horarios son tomados en consideración para recordarle al usuario que antes de una comida puede consultar a la aplicación si un alimento presenta interacción con algún medicamento.

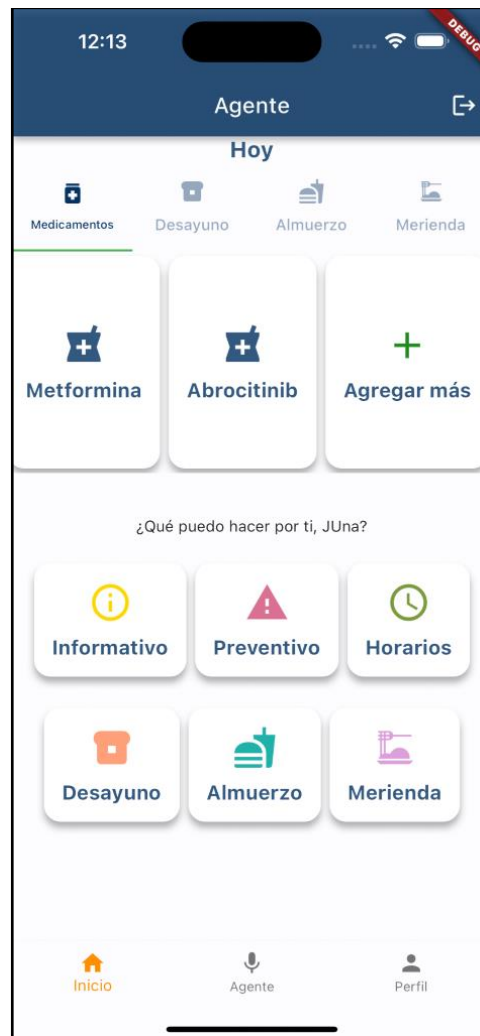
Figura 47 Autenticación del usuario.



- Pantalla principal

Una vez autenticado, el usuario ingresa a la aplicación, desde la cual podrá visualizar sus medicamentos personales (Figura 48).

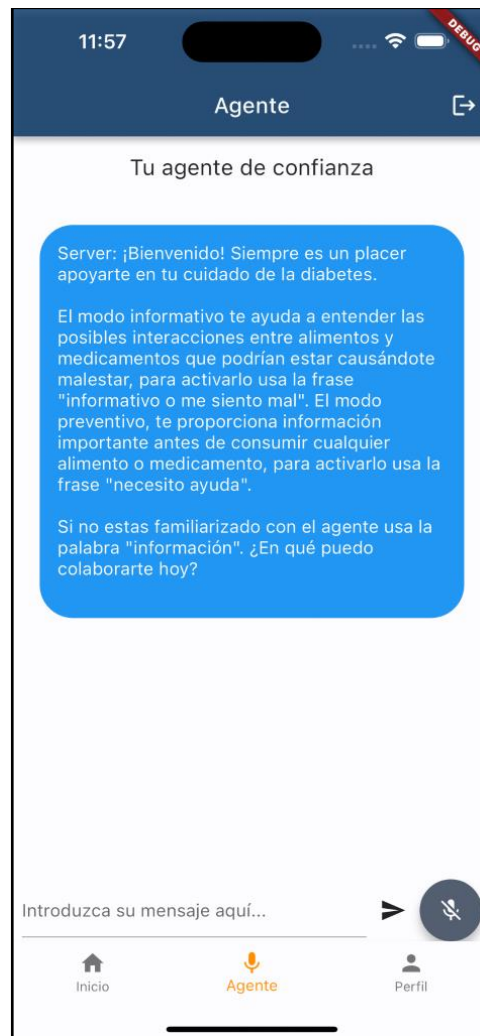
Figura 48 Autenticación del usuario.



- Solicitud y visualización de recomendaciones

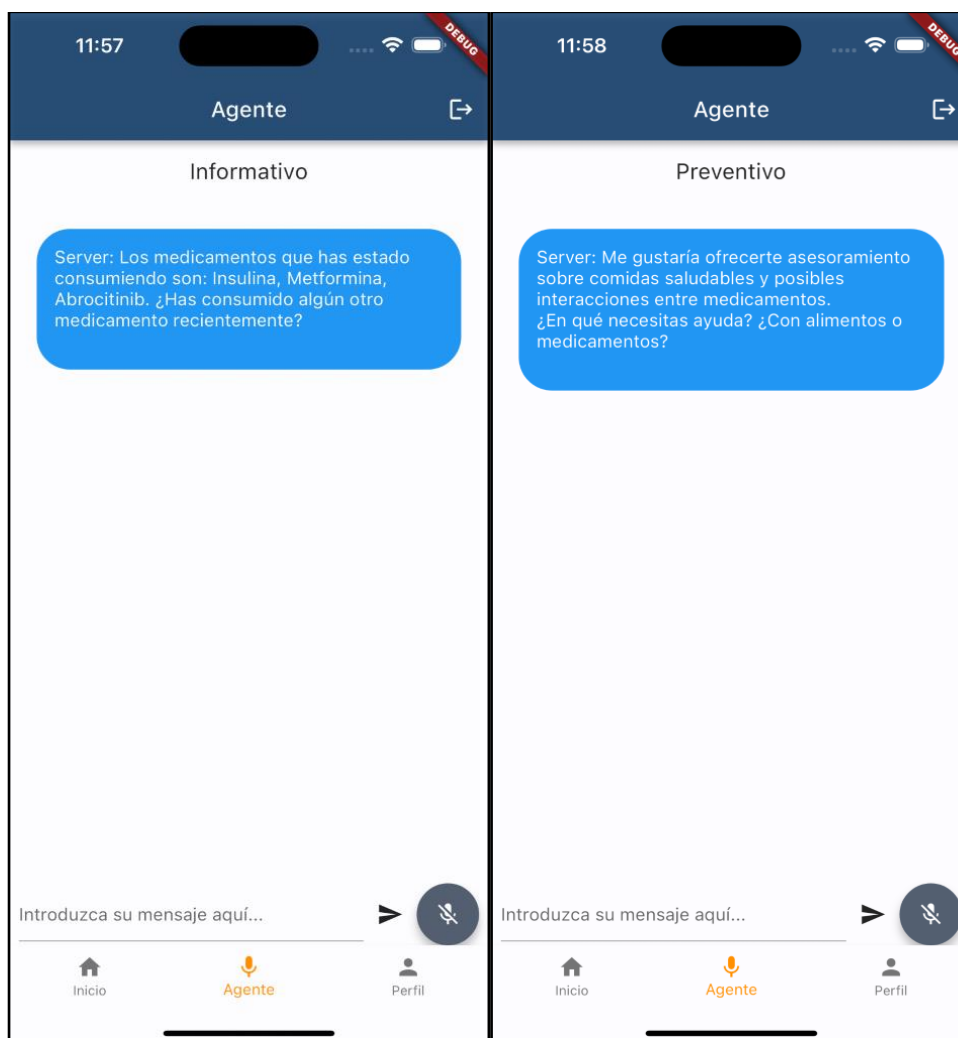
Una vez autenticados, los usuarios tienen la posibilidad de solicitar recomendaciones a través de la interfaz de chat. En el contexto actual, la funcionalidad de reconocimiento de voz del sistema se encuentra limitada exclusivamente a dispositivos que operan con el sistema iOS. Si ingresa al agente a través del icono 'Micrófono' será llevado a la parte como muestra en la Figura 49.

Figura 49 Autenticación del usuario.



Posteriormente, si el usuario desea ingresar al modo 'Preventivo' o modo 'Informativo', lo puede realizar desde la pantalla principal con los íconos correspondientes. Como se puede observar en la Figura 50.

Figura 50 Interfaz de usuario, agente conversacional: modo informativo y preventivo.

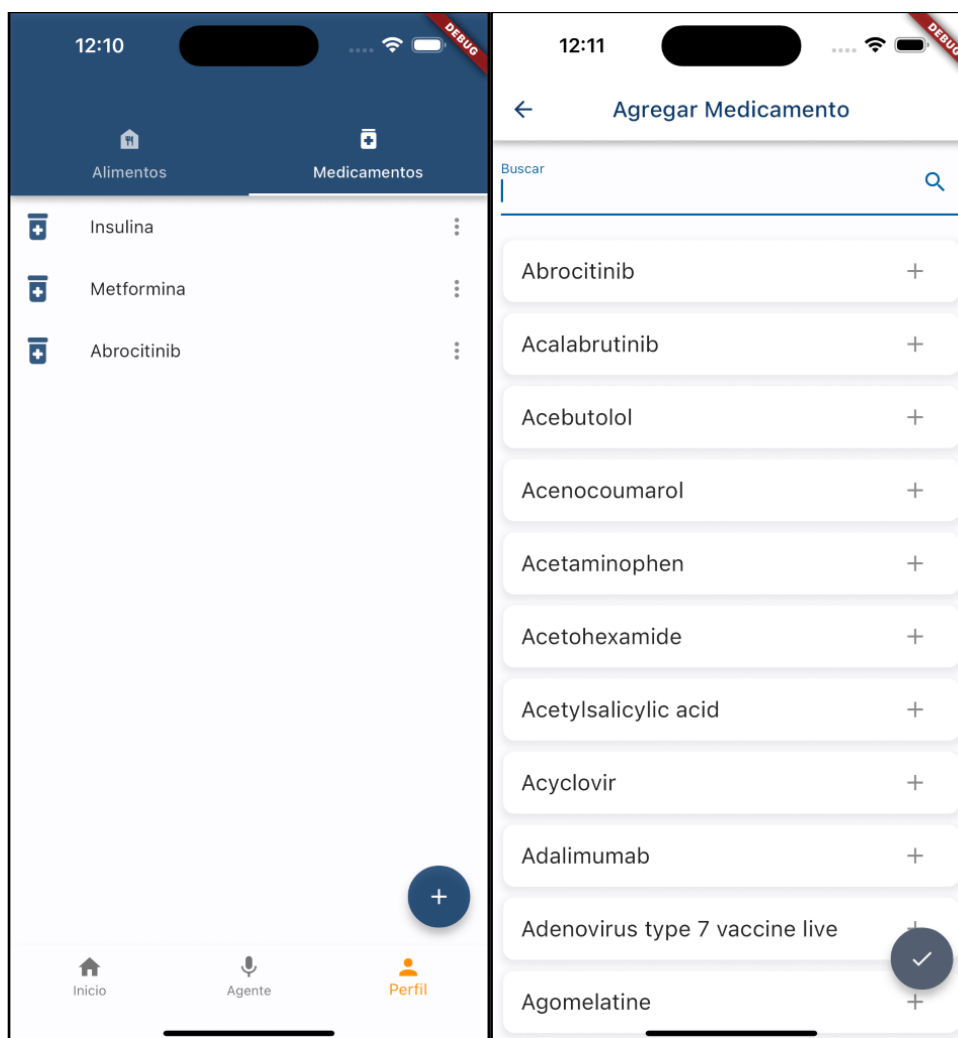


Una vez que el usuario interactúa con la aplicación, ésta presenta la información que pueden incluir contraindicaciones en el caso de que consulte sobre algún medicamento y este tenga interacción con el medicamento que lleva tomando, además las interacciones entre los fármacos-alimentos y la alternativa de alimento para evitar esa interacción. Esta información es presentada a través de la voz de la aplicación.

- Medicamentos

La Figura 51 muestra la opción para visualizar y agregar alimentos del usuario.

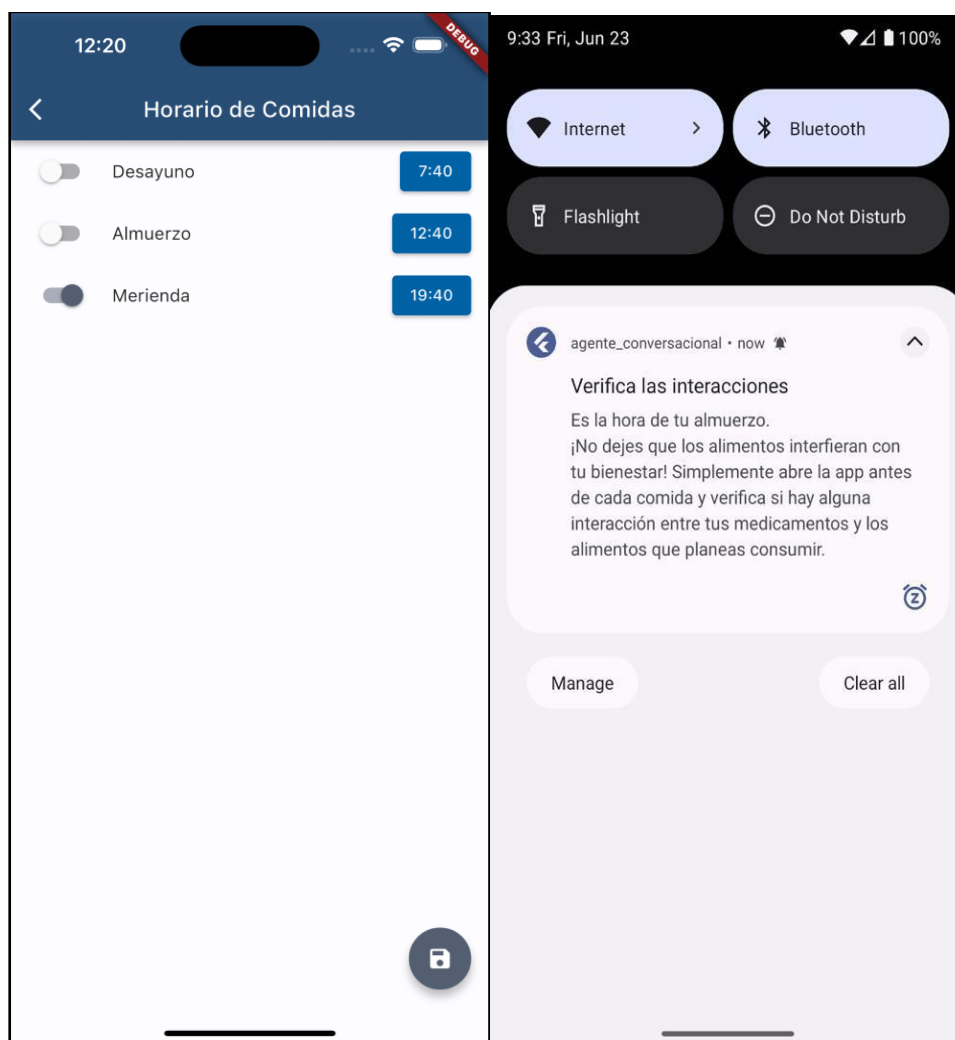
Figura 51 Interfaz de usuario, agente conversacional: registro de alimentos.



- Notificaciones

La funcionalidad de notificaciones de la aplicación permite mantener al usuario consciente de la existencia de una herramienta consultable, la cual puede proporcionar información valiosa sobre posibles interacciones entre los alimentos que se planea consumir y los medicamentos que se están tomando. Además, la aplicación ofrece la opción de personalizar las alertas, permitiendo al usuario activar o desactivar las notificaciones según sus preferencias, garantizando así que el sistema de notificaciones se ajuste a las necesidades individuales y evita la intrusión innecesaria. Se puede observar esto en la Figura 52.

Figura 52 Interfaz de usuario, notificaciones.

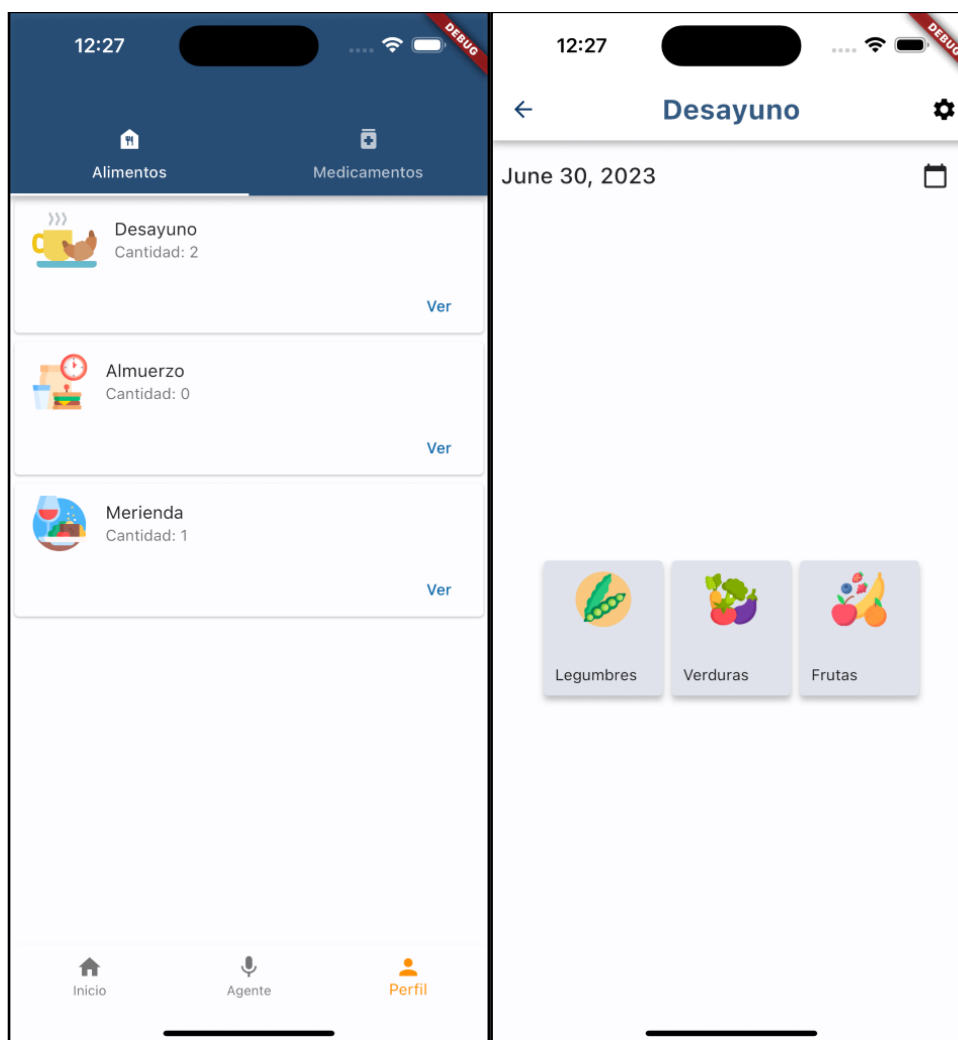


- Alimentos

La Figura 53 muestra una funcionalidad que permite a los usuarios registrar sus comidas, respondiendo a la tendencia observada de que los usuarios tienden a valorar la capacidad de hacer seguimiento a su ingesta alimentaria. Este componente, aunque útil en su estado actual, presenta posibilidades significativas para el desarrollo futuro. Se contempla la integración con el agente conversacional como una mejora a implementar en futuros trabajos, lo que simplificará y automatizará el proceso de consulta diaria.

En lugar de tener que interactuar manualmente con el agente cada día, el usuario podría simplemente referirse a los registros de alimentos previamente ingresados. Esta automatización se extendería no sólo a la alimentación, sino también al seguimiento de medicamentos, proporcionando al usuario un sistema integral de gestión de dieta y medicación. Al permitir el registro de comidas por días, esta funcionalidad brinda al usuario una mayor capacidad para personalizar y automatizar su experiencia con la aplicación.

Figura 53 Interfaz de usuario, comidas.



5.2. Limitaciones y desafíos del reconocimiento de voz en Android

A pesar de los beneficios y la versatilidad de estas bibliotecas, es crucial resaltar que, durante las pruebas de la aplicación, se identificó una limitación considerable. En particular, se observó que la entrada de audio en la plataforma Android no funciona de manera óptima al utilizar la biblioteca Speech To Text. Esta limitación plantea un desafío para la funcionalidad y la experiencia del usuario de la aplicación en dispositivos Android. Sin embargo, es importante destacar que este problema específico no se presentó en la plataforma iOS, donde el reconocimiento de voz funciona adecuadamente.

Por lo tanto, se requiere una investigación adicional y, posiblemente, un desarrollo más profundo para resolver este problema en dispositivos Android. El objetivo es garantizar que la aplicación pueda ofrecer una experiencia de usuario uniforme y de alta calidad en todas las plataformas, independientemente del sistema operativo del dispositivo. Esto es

fundamental para la visión de nuestro proyecto, que aspira a una amplia accesibilidad y funcionalidad de nuestra aplicación en una variedad de dispositivos y plataformas.

En el siguiente capítulo, se presentarán los resultados obtenidos a partir de la evaluación y el uso del aplicativo móvil con el agente conversacional implementado. Se analizará la usabilidad del aplicativo, se discutirá la experiencia de usuario y se plantearán posibles mejoras y recomendaciones.

Capítulo 6 – EVALUACIÓN Y RESULTADOS

Un sistema de recomendación basado en las interacciones fármaco-alimento y fármaco-fármaco es una herramienta valiosa dentro del campo farmacológico. El objetivo de este capítulo es realizar la evaluación del mismo y exponer los resultados asociados. La evaluación del sistema es de vital importancia para determinar su efectividad y utilidad para el usuario. Con el fin de obtener una visión completa y objetiva, se llevaron a cabo dos tipos de evaluaciones complementarias: una evaluación de usabilidad y una evaluación basada en el juicio de expertos.

6.1. Evaluación de usabilidad

La evaluación de usabilidad requiere de métodos no tradicionales, con el fin de proporcionar una imagen clara de la usabilidad del sistema. La evaluación tiene como objetivo comprender la experiencia de usuario (UX) con el sistema de recomendación.

La evaluación se realizó a través de un cuestionario de usabilidad de chatbot (CUQ), que en este contexto hace referencia al agente conversacional; se utilizó una muestra a conveniencia, es decir se seleccionaron participantes que se encuentran disponibles y dispuestos a participar en la evaluación. En algunos casos, hay necesidad de modificar los métodos de usabilidad con el fin de adaptarse a las características de los participantes o tecnologías. En este caso concreto, Holmes, S. et al. (2019), asegura que los cuestionarios de usabilidad tradicionales pueden no englobar todos los aspectos de la interfaz y que las métricas son más representativas si se miden aspectos de usabilidad estrechamente relacionados con los chatbots. Por este motivo, se hace uso de este tipo de cuestionario y las preguntas se han adaptado al contexto del agente conversacional; los evaluadores del agente fueron seleccionados a conveniencia y únicamente recibieron una inducción de los conceptos sobre: sistema, interfaz de usuario y agente conversacional.

6.1.1. Cuestionario de Usabilidad de Chatbot (CUQ)

El CUQ es un cuestionario diseñado para evaluar los aspectos positivos y negativos de un chatbot en base a los principios de UX proporcionados por la herramienta ALMA Chatbot Test. Estos principios incluyen la personalidad, la incorporación, la navegación, la comprensión, las respuestas, el manejo de errores y la inteligencia del chatbot (Martin, J. et al., 2017). El CUQ propuesto por Holmes, S. et al. (2019) consta de 16 preguntas que miden el acuerdo de los participantes para aspectos positivos y negativos del chatbot. La inclusión de preguntas negativas y positivas permite medir tanto los aspectos positivos como los negativos de la experiencia del usuario con el chatbot, lo que proporciona una imagen más completa de la usabilidad del chatbot.

Del conjunto de preguntas mencionado se escogieron 11, las cuales se adaptan a las necesidades y aspectos específicos (positivos y negativos) del agente conversacional en cuestión; cada pregunta se califica en una escala de Likert de cinco puntos, donde 'Totalmente de acuerdo' representa el valor más alto y 'Totalmente en desacuerdo' el más bajo. Además, es importante mencionar que el cuestionario se publicó en la web⁸. Fue completado por 26 estudiantes de la carrera de Nutrición y Dietética de la Universidad de Cuenca, con una participación de 22 mujeres y 4 hombres. Las preguntas utilizadas en el CUQ se encuentran en la Tabla 5.

Tabla 5 Cuestionario de usabilidad del sistema de recomendación.

Número de Pregunta	Pregunta
1	¿La interfaz es intuitiva y fácil de usar?
2	¿El agente conversacional no explicó bien su alcance y propósito?
3	¿El proceso de registro de información fue sencillo y sin complicaciones?
4	¿Es fácil confundirse al usar el sistema y el agente conversacional?
5	¿El agente conversacional entendió las solicitudes de manera correcta?
6	¿El agente conversacional no fue capaz de manejar las solicitudes de manera efectiva?
7	¿Las respuestas del agente conversacional fueron útiles y apropiadas?
8	¿El agente conversacional no fue capaz de mantener una conversación coherente?
9	¿El agente conversacional proporcionó información clara y concisa?
10	¿El agente conversacional no fue capaz de manejar errores de manera efectiva?
11	¿El agente conversacional fue capaz de proporcionar alternativas de consumo de alimentos?

Para obtener la puntuación final de la evaluación, los cálculos realizados fueron considerados con base en las 11 preguntas planteadas, siguiendo los pasos detallados a continuación:

1. Asignar una puntuación de 1 a 5 a cada pregunta en función de la escala de Likert.
2. Calcular la suma de todas las preguntas positivas (preguntas impares).
3. Calcular la suma de todas las preguntas negativas (preguntas pares).
4. Restar 6 al total obtenido en el paso 2.
5. Restar el total obtenido en el paso 3 de 25.
6. Sumar las puntuaciones obtenidas en los pasos 4 y 5.
7. Dividir la puntuación obtenida en el paso 6 por 44 y multiplicar el resultado por 100 para obtener un CUQ sobre 100.

La resta de 6 en las puntuaciones de las preguntas positivas y la resta del total de las puntuaciones de las preguntas negativas de 25 se realizan para ajustar las puntuaciones y garantizar que las preguntas tanto positivas como negativas contribuyen de manera proporcional al CUQ.

En total, la puntuación del CUQ puede variar entre 11 y 55, esto si partimos del cuestionario planteado. Si nos enfocamos en las preguntas positivas, la suma total de las puntuaciones oscila entre 6 y 30, mientras que, para las preguntas negativas, la suma total de las puntuaciones puede variar entre 5 y 25. Estos rangos proporcionan una amplia distribución para evaluar la usabilidad y eficacia del agente conversacional de manera efectiva.

Por tanto, la finalidad de este proceso es realizar una evaluación sistemática y cuantitativa del rendimiento del sistema, proporcionando una visión detallada de sus aspectos positivos y negativos. La aplicación de este enfoque permitirá identificar áreas de mejora y facilitará la toma de decisiones para futuras optimizaciones y mejoras en el sistema.

En cuanto a los resultados, se consideraron los aspectos positivos que hacen referencias a las preguntas impares de la Tabla 5, mientras que para los aspectos negativos del chatbot se consideraron las preguntas pares; también se realizó un análisis de las preguntas de cada grupo, con base a sus resultados.

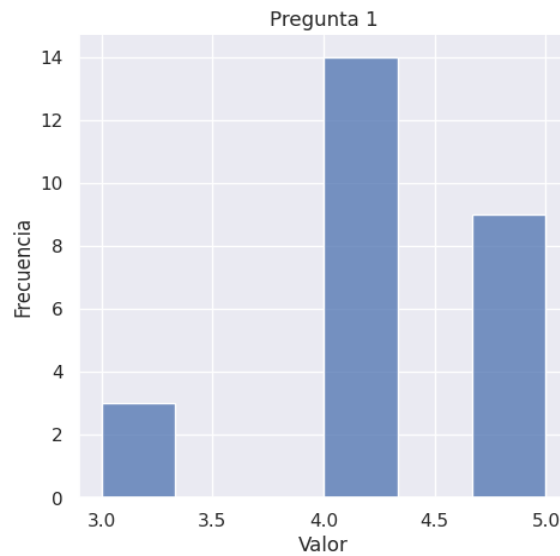
- Aspectos positivos

Para el análisis de los aspectos positivos del chatbot, se abarca diferentes aspectos relacionados con la interfaz, la comprensión, la utilidad y la claridad del agente conversacional. Los resultados de estas preguntas proporcionan una visión específica de la experiencia de los usuarios. A partir de ello, se detallan los hallazgos más relevantes de cada una.

En términos generales, los usuarios coincidieron en que la interfaz del agente conversacional resulta intuitiva y de fácil uso, según la pregunta 1. Esto indica que la presentación y la

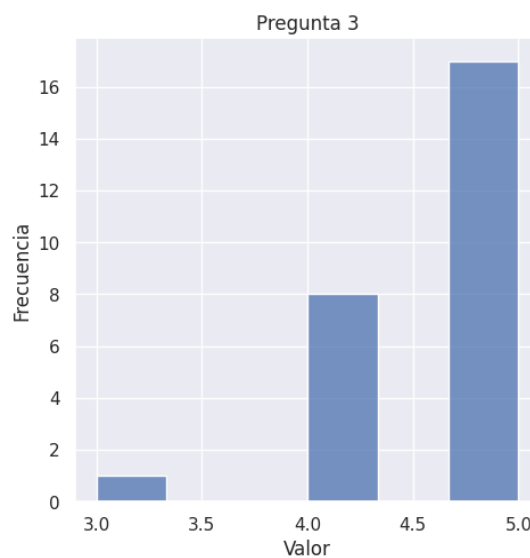
navegación del sistema son efectivas, lo cual es muy positivo en cuanto a su usabilidad (Figura 54).

Figura 54 Resultados de la pregunta 1 de la evaluación de usabilidad.



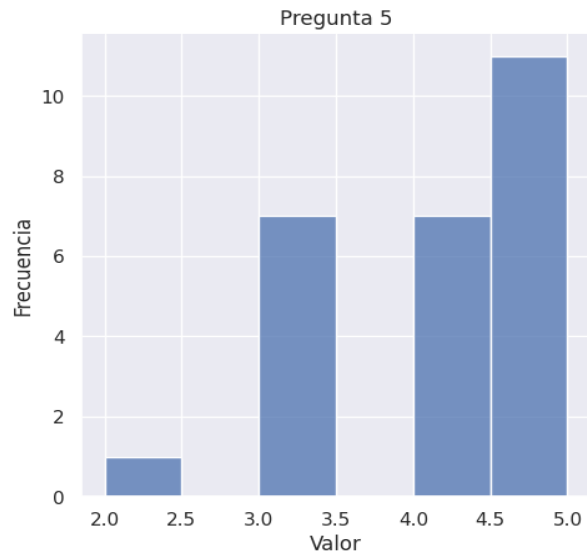
La pregunta 3 indica que la mayoría de los usuarios consideraron que el proceso de registro de información fue sencillo y sin complicaciones (Figura 55). Esta percepción positiva indica que el agente conversacional proporcionó una experiencia fluida y eficiente durante el proceso de registro. Un proceso de registro de información sencillo y sin complicaciones es crucial para garantizar una experiencia positiva del usuario, ya que reduce la fricción y facilita el acceso a los servicios del chatbot.

Figura 55 Resultados de la pregunta 3 de la evaluación de usabilidad.



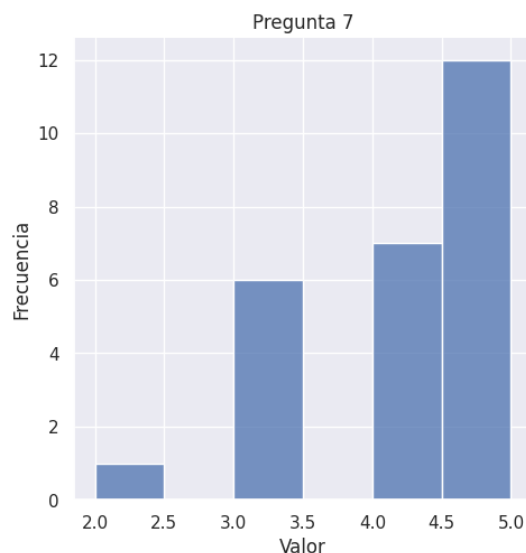
Los evaluadores expresaron en la pregunta 5 opiniones variadas sobre si el agente conversacional comprendió las solicitudes de manera correcta (Figura 56). Esto sugiere que hay margen de mejora en la capacidad del agente para interpretar y responder adecuadamente a las consultas de los usuarios.

Figura 56 Resultados de la pregunta 5 de la evaluación de usabilidad.



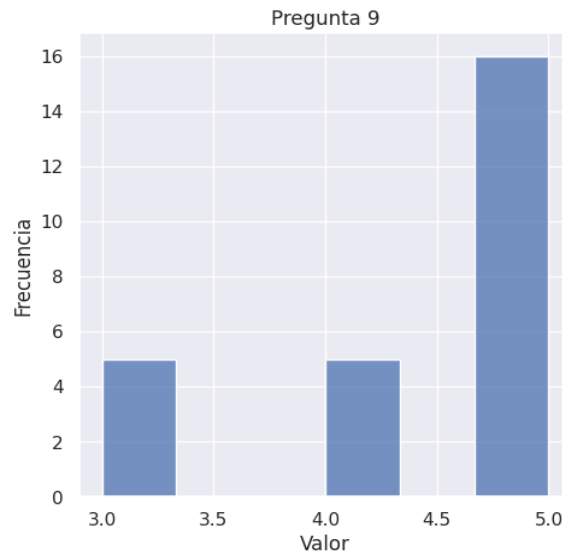
En cuanto a la pregunta 7, que abarca la utilidad y adecuación de las respuestas del agente conversacional, los resultados indican una mezcla de opiniones. Algunos usuarios consideraron que las respuestas fueron útiles y apropiadas, mientras que otros tuvieron una experiencia menos satisfactoria (Figura 57). Es importante analizar las respuestas individuales para identificar patrones específicos que puedan guiar mejoras.

Figura 57 Resultados de la pregunta 7 de la evaluación de usabilidad.



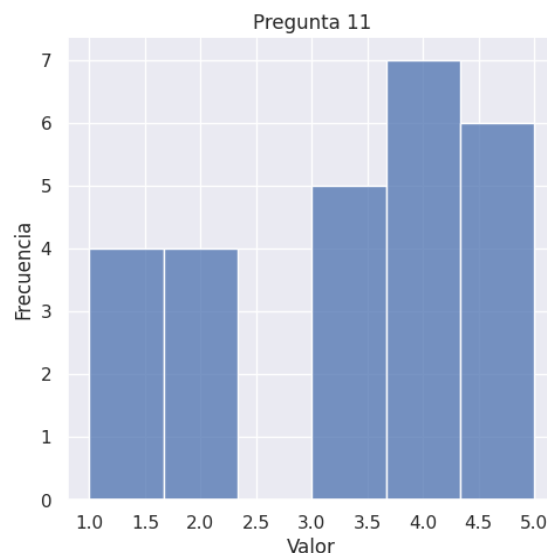
En relación a la claridad y concisión de la información proporcionada por el agente conversacional, planteada en la pregunta 9, la mayoría de los usuarios indicaron que las respuestas fueron claras y concisas (Figura 58). Esto es un indicador positivo, ya que la comunicación efectiva es esencial para garantizar una experiencia satisfactoria.

Figura 58 Resultados de la pregunta 9 de la evaluación de usabilidad.



En cuanto a la capacidad del agente conversacional para proporcionar alternativas de consumo de alimentos (pregunta 11), los resultados mostraron una variedad de respuestas. Algunos usuarios consideraron que el agente fue capaz de proporcionar alternativas, mientras que otros no tuvieron una experiencia tan positiva (Figura 59). Esto señala la importancia de mejorar la capacidad del agente para brindar opciones relevantes y adecuadas.

Figura 59 Resultados de la pregunta 11 de la evaluación de usabilidad.

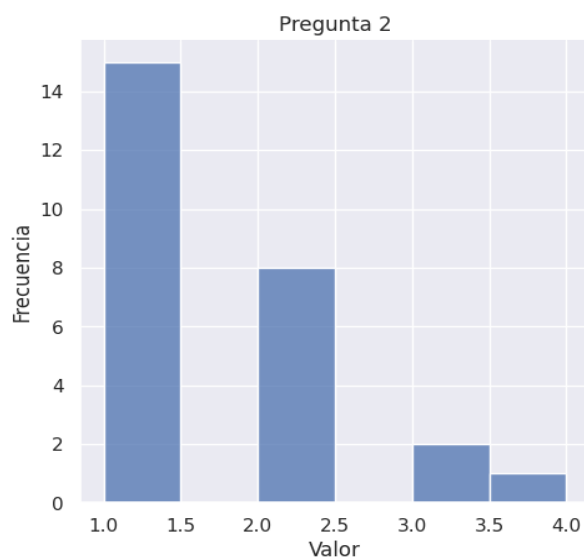


- Aspectos Negativos

Los aspectos negativos corresponden a las desventajas y áreas de mejora del agente, centrándose en áreas críticas de la experiencia de los usuarios con el agente conversacional. Con base en esto, se presentan a continuación los resultados más significativos de cada pregunta.

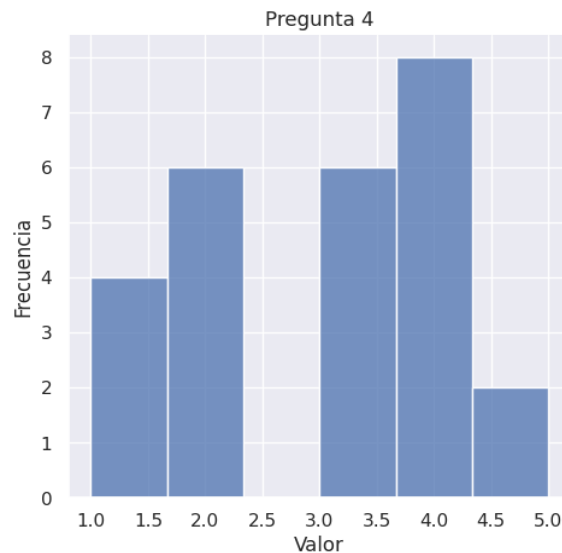
Los resultados obtenidos en la pregunta 2 indican que hubo cierta falta de claridad en este aspecto. Algunos usuarios expresaron una calificación baja, lo cual apunta que existieron dificultades para comprender completamente el alcance y propósito del agente conversacional (Figura 60). Esta falta de claridad puede generar confusión y afectar la percepción de los usuarios sobre la funcionalidad del chatbot. Para mejorar la experiencia del usuario, es necesario revisar y reforzar la forma en que se comunica el alcance y propósito del agente conversacional.

Figura 60 Resultados de la pregunta 2 de la evaluación de usabilidad.



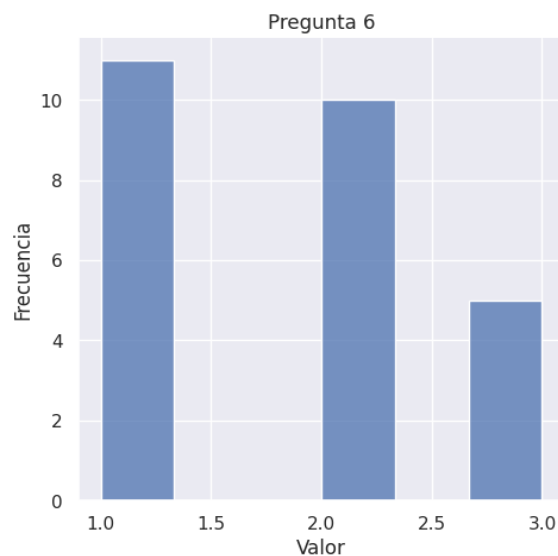
En cuanto a la pregunta 4, los resultados reflejan una variedad de respuestas. Algunos usuarios indicaron que experimentaron confusión al utilizar el sistema y el agente conversacional (Figura 61). Esto sugiere que puede haber aspectos de la interfaz o del flujo de conversación que necesitan mejorarse para reducir la confusión y mejorar la experiencia del usuario.

Figura 61 Resultados de la pregunta 4 de la evaluación de usabilidad.



En relación a la pregunta 6, los resultados indican que la mayoría de los usuarios consideraron que el agente conversacional fue capaz de manejar las solicitudes de manera efectiva (Figura 62). Esto resalta la capacidad del agente para comprender y responder adecuadamente a las solicitudes de los usuarios, lo cual es esencial para brindar una experiencia útil y satisfactoria.

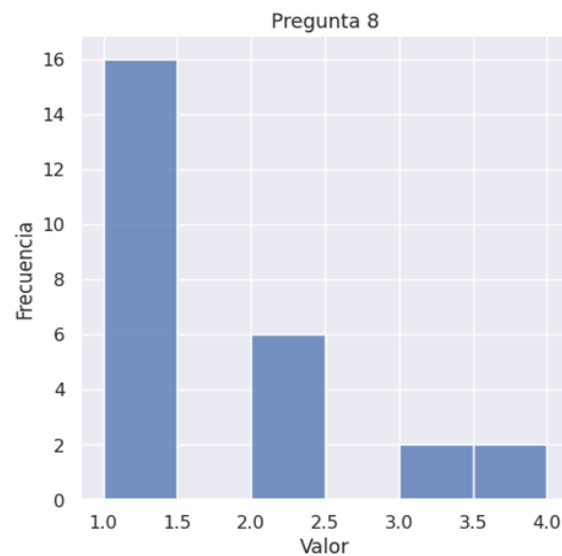
Figura 62 Resultados de la pregunta 6 de la evaluación de usabilidad.



En la pregunta 8, los resultados advierten que el agente conversacional fue capaz de mantener una conversación coherente de manera efectiva. Esto indica que el chatbot ha sido diseñado de tal manera que le permite comprender el contexto de las conversaciones y generar respuestas coherentes y relevantes (Figura 63). La capacidad del agente para

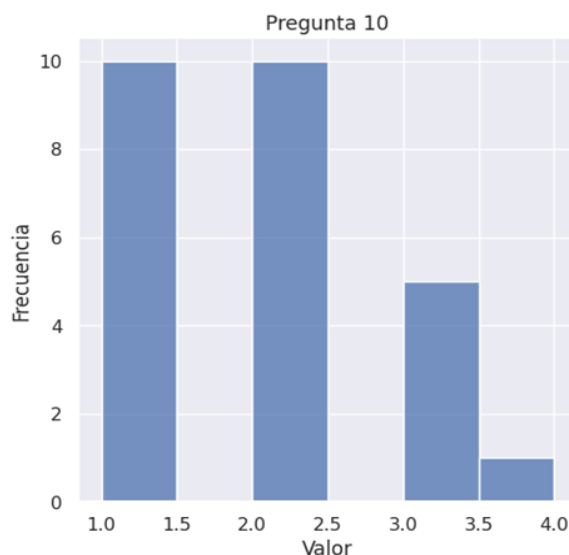
mantener la coherencia en las interacciones es fundamental para proporcionar una experiencia satisfactoria y agradable a los usuarios. Es importante seguir monitoreando y mejorando constantemente esta capacidad para garantizar una comunicación fluida y natural en futuras interacciones con el chatbot.

Figura 63 Resultados de la pregunta 8 de la evaluación de usabilidad.



En relación a la pregunta 10, la mayoría de los usuarios consideraron que el agente conversacional fue capaz de manejar errores de manera efectiva (Figura 64). Esto demuestra que el chatbot ha sido diseñado con mecanismos eficientes para reconocer y recuperarse de los errores, brindando respuestas útiles y claras en situaciones de error. Esta capacidad del agente conversacional contribuye positivamente a la experiencia del usuario, generando confianza y evitando posibles frustraciones. Sin embargo, siempre es importante seguir mejorando y perfeccionando la capacidad del agente para garantizar una interacción fluida y libre de errores.

Figura 64 Resultados de la pregunta 10 de la evaluación de usabilidad.

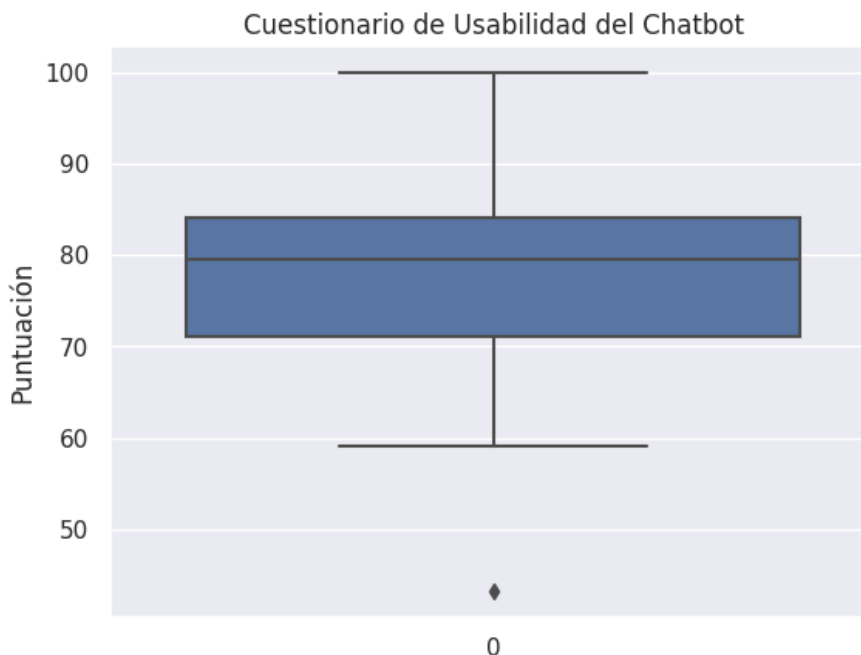


En resumen, los resultados positivos y negativos de la usabilidad del agente conversacional, proporcionan una visión específica de la experiencia de los usuarios con el agente conversacional, identifican las áreas de mejora específicas, como la comprensión de las solicitudes, la utilidad de las respuestas y la capacidad de proporcionar alternativas relevantes; también permitirá realizar ajustes o mejoras en el sistema para brindar una experiencia más efectiva y satisfactoria al usuario.

Además, destacan áreas críticas que requieren atención y mejoras en el agente conversacional. Al abordar estas deficiencias, se puede mejorar la experiencia del usuario y aumentar la eficacia y la satisfacción al interactuar con el agente. Es importante analizar en detalle las respuestas individuales y tomar medidas específicas para abordar las deficiencias identificadas en cada pregunta.

Tras analizar detenidamente tanto los aspectos positivos como los negativos, se han obtenido los resultados de la evaluación del agente conversacional. La media resultante es de 77.27 +/- 11.62, lo que indica que, en general, los usuarios han manifestado estar de acuerdo o bastante de acuerdo con las afirmaciones planteadas en el cuestionario de usabilidad del chatbot. Cabe destacar que se ha registrado una puntuación máxima de 100 y una mínima de 43.18 (Figura 65).

Figura 65 Resultados del cuestionario de usabilidad del chatbot.



6.2. Evaluación del juicio del experto

El juicio de expertos es una técnica que se utiliza para recopilar información y conocimiento de individuos que poseen experiencia o son expertos en un campo específico, con el fin de evaluar cantidades o parámetros que son desconocidos o inciertos (Ryan, J. et al., 2012). Este método implica la solicitud, codificación y combinación del conocimiento de los expertos y se utiliza cuando los datos cuantitativos no son suficientes o su calidad es dudosa para obtener resultados estadísticos precisos; también, se utiliza para combinar las perspectivas de los expertos y obtener así una visión combinada que representa la opinión colectiva de los estos.

En el contexto del agente conversacional, el juicio de los expertos se ha utilizado para verificar las respuestas del agente frente a las posibles interacciones que puedan darse, a partir de la elaboración de escenarios (Anexo A-D). Dos profesionales de la salud diseñaron 4 escenarios que incluyen tanto escenarios preventivos e informativos sobre las interacciones.

Para comprobar las respuestas proporcionadas por el agente conversacional en situaciones del mundo real, se contó con la participación de cuatro expertos con los siguientes perfiles: un químico farmacéutico con un doctorado en manejo de interacciones, un médico con experiencia en el tratamiento de pacientes con diabetes, un nutricionista con experiencia en la gestión de interacciones entre alimentos y fármacos, y un médico con experiencia diaria en consultas médicas. Además, se involucraron dos evaluadores que compararon las respuestas

de los expertos con las del sistema de recomendación mediante un cuestionario diseñado en base a los escenarios planteados. A continuación, se presentan los resultados obtenidos para cada escenario:

6.2.1. Escenario 1

En el escenario 1, los evaluadores, otorgaron calificaciones positivas al sistema de recomendación del chatbot basado en interacciones fármaco-fármaco y fármaco-nutriente.

Ambos evaluadores estuvieron de acuerdo en que el sistema identificó correctamente las posibles interacciones entre la Metformina y la Simvastatina, así como entre la Simvastatina y la Toronja. Esto demuestra la capacidad del sistema para reconocer y advertir sobre estas interacciones potenciales, siendo crucial para garantizar la seguridad y la eficacia del tratamiento farmacológico.

En cuanto a las recomendaciones proporcionadas por el sistema, los evaluadores mostraron cierto grado de conformidad, no obstante, también aconsejan una posibilidad de mejora. Esto indica que consideran que las recomendaciones son útiles y adecuadas, pero puede haber espacio de mejora ya que el agente arroja una idea general y no abarca la diversidad de opiniones por parte de los expertos respecto a los efectos secundarios de las interacciones de los medicamentos presentes en este escenario.

6.2.2. Escenario 2

Se obtuvieron calificaciones positivas en general; la puntuación obtenida refleja un alto grado de acuerdo en cuanto a la capacidad del sistema de recomendación para identificar interacciones.

En primer lugar, están totalmente de acuerdo en que el sistema identificó correctamente las posibles interacciones entre la Insulina y el Enalapril, así como entre el Enalapril y algunos alimentos mencionados. Esto indica que el sistema fue capaz de reconocer y advertir sobre las interacciones potenciales, lo cual es fundamental para evitar problemas de seguridad y maximizar la eficacia del tratamiento.

Además, aunque con una calificación ligeramente más baja, consideraron que las recomendaciones proporcionadas por el sistema en cuanto al consumo de alimentos son en su mayoría útiles y adecuadas. Esto sugiere que las recomendaciones ofrecidas brindan información valiosa sobre las alternativas en el consumo de alimentos que podrían ser beneficiosas en el contexto de la interacción fármaco-nutriente, con las observaciones señaladas en el escenario 1.

6.2.3. Escenario 3

Para este escenario se obtuvieron calificaciones mixtas por parte de los evaluadores. Los puntajes reflejan cierto grado de incertidumbre y discrepancia en la evaluación del sistema de recomendación del chatbot.

En cuanto a la identificación de posibles interacciones entre los medicamentos planteados y la manzana, mostraron una calificación neutral, lo que sugiere no estar seguros o convencidos de la capacidad del sistema para reconocer esta interacción específica; ya que, podría haber espacio para mejorar la detección de interacciones relacionadas con la manzana y los medicamentos.

Por otro lado, en la identificación de posibles interacciones entre la Cerveza y algunos de los medicamentos, los evaluadores estuvieron de acuerdo en cierta medida. Lo que implica que reconocen en parte la capacidad del sistema para detectar y advertir sobre esta interacción, aunque aún pueden existir algunas dudas o reservas en su evaluación.

En cuanto a las recomendaciones proporcionadas por el sistema, los evaluadores consideraron que son en su mayoría útiles y adecuadas, pero también señalaron la posibilidad de mejoras adicionales. Esto propone que las recomendaciones ofrecidas por el sistema pueden brindar cierto valor, pero aún hay aspectos que podrían ser refinados para mejorar su utilidad y precisión.

6.2.4. Escenario 4

En este escenario se indica un acuerdo general en cuanto a la capacidad del sistema de recomendación, en relación a la identificación de las eventuales interacciones entre los medicamentos citados, los evaluadores estuvieron de acuerdo en que el sistema pudo identificar correctamente estas interacciones. Esto refleja que el sistema es capaz de reconocer y advertir sobre las interacciones potenciales entre los medicamentos, lo cual es esencial para la seguridad y efectividad del tratamiento. Asimismo, los evaluadores coincidieron en que las recomendaciones proporcionadas por el sistema son útiles y adecuadas. Esto indica que las recomendaciones ofrecidas brindan información valiosa y pertinente.

En resumen, los escenarios evaluados muestran que el sistema de recomendación tiene la capacidad para identificar interacciones fármaco-fármaco y fármaco-alimento, y para proporcionar recomendaciones relevantes. Si bien existen áreas donde se pueden hacer mejoras adicionales, en general, los evaluadores reconocieron la utilidad y precisión del

sistema en la mayoría de los escenarios. Estos resultados respaldan la efectividad del sistema en brindar orientación segura y efectiva al usuario.

Capítulo 7 – DISCUSIÓN

En el presente apartado, se discutirán los resultados obtenidos en la evaluación de usabilidad del chatbot y el juicio del experto basado en las interacciones fármaco-fármaco y fármaco-nutriente, así como sus implicaciones y recomendaciones para mejorar la experiencia del usuario.

Los resultados de la evaluación indican una calificación promedio de 77.27, ponderada sobre 100, lo cual refleja una percepción generalmente positiva de los usuarios hacia la usabilidad del chatbot. Los puntajes obtenidos en cada pregunta muestran que, en su mayoría, los usuarios se encuentran de acuerdo o totalmente de acuerdo con las afirmaciones evaluadas.

Entre las fortalezas identificadas, se destaca la percepción positiva en cuanto a la intuitividad y facilidad de uso de la interfaz, así como la claridad y concisión de las respuestas proporcionadas por el agente conversacional. Estos resultados sugieren que el diseño y la funcionalidad del chatbot han logrado transmitir de manera efectiva la información al usuario.

Además, los resultados obtenidos en los cuatro escenarios evaluados respaldan la efectividad del sistema de recomendación del chatbot en la identificación de interacciones farmacológicas y la provisión de recomendaciones útiles. A pesar de contar con áreas de mejora en los escenarios, estos hallazgos enfatizan la importancia de contar con sistemas precisos y confiables para brindar orientación segura y efectiva a los usuarios en relación con las interacciones entre medicamentos y alimentos. También, destacan la relevancia de la evaluación del juicio del experto como un componente crítico para validar y mejorar los sistemas de recomendación. Futuras mejoras podrían incluir la incorporación de más bases de datos, la optimización de los algoritmos de recomendación y la personalización de las recomendaciones para aumentar aún más la utilidad y la precisión del sistema en diferentes escenarios.

No obstante, los resultados también revelan áreas de mejora. Específicamente, se han identificado aspectos relacionados con la explicación del alcance y propósito del agente conversacional, la capacidad para manejar errores de manera efectiva y mantener una conversación coherente. Estas áreas señalan la necesidad de realizar ajustes y mejoras para optimizar la usabilidad del agente conversacional.

Durante la evaluación, se recopilaron comentarios y sugerencias de los usuarios. Estos comentarios proporcionan información adicional y perspectivas importantes sobre la usabilidad del chatbot. Algunos usuarios expresaron la necesidad de una mayor personalización en las respuestas o la inclusión de más alternativas de consumo de

alimentos. Estos comentarios pueden ser utilizados como guía para realizar ajustes específicos y satisfacer mejor las necesidades y expectativas de los usuarios.

7.1. Implicaciones y recomendaciones

Basado en los resultados y el análisis realizado, se recomienda lo siguiente:

Mejorar la explicación del alcance y propósito del agente conversacional: Proporcionar una introducción clara y concisa sobre las capacidades y limitaciones del chatbot para que los usuarios tengan una comprensión clara de lo que pueden esperar.

Fortalecer la capacidad para manejar errores: Implementar mecanismos efectivos para detectar y manejar errores de manera proactiva, brindando respuestas claras y útiles para corregir o aclarar la información proporcionada.

Mantener una conversación coherente: Optimizar el procesamiento del lenguaje natural del agente conversacional para mejorar su capacidad de mantener una conversación fluida y coherente, evitando respuestas confusas o contradictorias.

Personalización de respuestas: Considerar la posibilidad de incorporar elementos de personalización en las respuestas del chatbot, teniendo en cuenta las preferencias individuales de los usuarios para ofrecer una experiencia más adaptada a sus necesidades.

Estas recomendaciones proporcionan una guía inicial para abordar las áreas de mejora identificadas y optimizar el agente conversacional en todas sus ámbitos y funcionalidades. Se sugiere llevar a cabo pruebas adicionales y recopilar más comentarios de los usuarios para continuar iterando y mejorando la experiencia de usuario en futuras versiones del chatbot.

7.2. Limitaciones del estudio

Es importante tener en cuenta las limitaciones de este trabajo, como el tamaño de la muestra y la representatividad de los participantes. Estas limitaciones pueden influir en la generalización de los resultados obtenidos. Se recomienda realizar estudios más amplios y diversificados en el futuro para obtener una visión más completa de la usabilidad del chatbot.

En cuanto a la aplicación, se ha observado que la configuración de las fechas de nacimiento requiere de una selección más sencilla para los usuarios. El actual proceso de agregar la fecha de nacimiento, implica varios pasos y se podría buscar una solución que los simplifique para facilitar esta tarea.

Otra consideración en términos de programación es la necesidad de ofrecer una interfaz de usuario que pueda indicar los pasos necesarios cuando el usuario ingresa un medicamento

por primera vez. Es importante guiarlo en las actividades siguientes, especialmente cuando usa la aplicación por primera vez. Además, se ha identificado que al agregar o modificar sus medicamentos, debe refrescar manualmente la aplicación desplazándose hacia abajo. Esta funcionalidad aún requiere ser controlada y mejorada.

Un aspecto a mencionar en la plataforma Android, fue la interacción con la voz, ya que, en ocasiones no reconoce la voz del usuario. Después de investigar en la documentación oficial de Flutter, se ha constatado que la biblioteca de reconocimiento de voz utilizada, Speech to Text, suele presentar estas situaciones en los dispositivos Android, debido a la dependencia de la versión del sistema operativo y del hardware del dispositivo. Dicha limitación se ha identificado durante las pruebas correspondientes. Sin embargo, en el caso de dispositivos iOS, la funcionalidad ha respondido de manera satisfactoria. Cabe recalcar que, independiente del sistema operativo, es necesario trabajar en el control de mensajes para evitar respuestas de bienvenida cuando el usuario abandona la aplicación.

En cuanto al agente, las evaluaciones han revelado aspectos importantes que pueden mejorar la interacción de la interfaz, como dar una funcionalidad extra a los botones de alimentos, pues solo tiene la posibilidad de registrarlos; también, se debe considerar la interacción con el agente, ya que, el usuario actualmente las preguntas del mismo se centran específicamente en el consumo de alimentos, y aún no se ha implementado preguntas relacionadas al contexto de las bebidas, por lo que es importante desarrollar esta funcionalidad.

Las alternativas de consumo de alimentos proporcionadas por el sistema, utilizando el algoritmo de filtrado basado en el contenido son consideradas importantes para optimizar el tiempo en el proceso de búsqueda, ya que, en las pruebas realizadas se evidenció que el tiempo necesario para consultar los datos del usuario y compararlos con los alimentos oscila entre 10 y 30 segundos. Sin embargo, este tiempo excede el rango considerado aceptable por Dialogflow, que solo espera 5 segundos para una respuesta adecuada. Por lo tanto, es necesario explorar diferentes estrategias con el fin de lograr una integración más eficiente entre este componente y la plataforma.

En resumen, los resultados de la evaluación de usabilidad del chatbot indican una percepción generalmente positiva por parte de los usuarios. Sin embargo, se han identificado áreas de mejora que requieren atención y ajustes. Al implementar las recomendaciones mencionadas, es posible mejorar la usabilidad del chatbot y proporcionar a los usuarios una experiencia más intuitiva y satisfactoria en su interacción con el sistema de recomendación basado en el chatbot.

Capítulo 8 – CONCLUSIONES

Alineados a los objetivos planteados, se describen las conclusiones de este trabajo:

Se desarrolló un agente conversacional, mismo que facilitará el consumo de medicamentos por los adultos mayores, tomando en cuenta diversas interacciones, entre ellas: fármaco-fármaco y fármaco-alimento.

- Diseñar una base de datos que permita almacenar información sobre fármacos, alimentos e información personal del usuario (adulto mayor).

La implementación de la base de datos, que incluye información relevante sobre las interacciones fármaco-fármaco y fármaco-alimento, así como las características de los alimentos, como su composición nutricional y su posición en la pirámide alimenticia. Además, esta implementación permitirá que, en futuras mejoras, el agente conversacional sea utilizado con fines médicos, adaptando la base de datos a las necesidades específicas de cada país y usuario. De esta manera, se garantiza que el agente pueda ofrecer recomendaciones adecuadas y precisas, respaldadas por una base de datos sólida y actualizada.

- Generar un algoritmo de recomendación de alternativas de consumo de alimentos para evitar interacciones entre fármacos y alimentos.

A través del agente conversacional diseñado, se ha desarrollado un algoritmo que brinda información personalizada a los adultos mayores sobre las interacciones entre fármacos y alimentos. Este algoritmo no solo proporciona información sobre las interacciones específicas, sino que también ofrece alternativas de consumo de alimentos utilizando el filtrado basado en el contenido.

El algoritmo brinda recomendaciones de alimentos de manera preventiva, teniendo en cuenta las posibles interacciones negativas entre los alimentos y los medicamentos, con el objetivo de mejorar la comprensión de los efectos colaterales por parte de los adultos mayores al ingerirlos conjuntamente.

- Implementar un prototipo de agente conversacional dentro de una aplicación móvil, para realizar las recomendaciones de las interacciones del fármaco con el paciente.

Finalmente, mediante el desarrollo de un aplicativo móvil, se integró el agente conversacional. Esta implementación se ha llevado a cabo para proporcionar una solución tecnológica que mejore la comunicación y la experiencia del usuario permite al usuario adulto mayor, conocer las diversas recomendaciones de los fármacos de manera personalizada.

La incorporación del agente conversacional en la aplicación móvil no solo permite una interacción más intuitiva y accesible para el usuario, sino que también le brinda la opción de recibir información mediante voz. Esto facilita que el usuario pueda realizar consultas y recibir recomendaciones de manera sencilla y conveniente, sin la necesidad de leer textos largos o navegar a través de múltiples opciones.

La implementación de este prototipo de agente conversacional en una aplicación móvil también ofrece la ventaja de estar disponible en cualquier momento y lugar, lo que facilita el acceso a la información y las recomendaciones por parte del paciente. Asimismo, la aplicación móvil puede enviar notificaciones y recordatorios para ayudar al paciente a evitar posibles interacciones perjudiciales.

8.1. Trabajos futuros

En cuanto a los trabajos futuros, es recomendable considerar la inclusión de mayor información en la base de datos utilizada, para lograr resultados más precisos y contextualizados. La adición de datos, como composición nutricional detallada de alimentos específicos y detalles regionales, permitiría obtener información más precisa y personalizada para los usuarios. Como una mejora adicional, se sugiere contextualizar la aplicación hacia frutas y alimentos propios de Ecuador. En la versión actual, se ha utilizado una base de datos con alimentos generales para realizar pruebas y evaluaciones. Sin embargo, al considerar la diversidad y riqueza de los alimentos ecuatorianos, sería beneficioso ampliar la base de datos para incluir frutas y alimentos típicos de la región. Esto permitiría brindar recomendaciones más relevantes y adaptadas a la dieta local, fomentando así una experiencia más enriquecedora para los usuarios en el contexto ecuatoriano.

Además, es importante destacar que, para la implementación de trabajos futuros, se deberá realizar una exhaustiva recopilación de datos provenientes de fuentes confiables y actualizadas, con el objetivo de garantizar la exactitud y relevancia de la información proporcionada por la aplicación. La colaboración con expertos en nutrición y la validación de la información recopilada también serán aspectos fundamentales a considerar.

Para el algoritmo de filtrado basado en contenido utilizado en la aplicación, se identifican diversas alternativas que podrían mejorar su rendimiento. Una posible estrategia consiste en analizar y optimizar el proceso de comparación de datos, así como implementar una estructura de datos más eficiente para agilizar la búsqueda. Además, se sugiere considerar técnicas de indexación o caché para acelerar las consultas repetitivas.

El preprocesamiento de datos puede resultar beneficioso, mediante la reducción de dimensionalidad o el agrupamiento de alimentos similares, lo que disminuiría el espacio de búsqueda y mejoraría la velocidad de las consultas. Asimismo, la implementación de un sistema de almacenamiento en caché de resultados previos evitaría cálculos repetidos y contribuiría a una respuesta más rápida.

Otra mejora potencial en la aplicación consiste en generar dos perfiles distintos: uno relacionado a usuarios y otra enfocada en profesionales de la salud, debido a los resultados de las evaluaciones realizadas por ellos y en la comparación de sus respuestas con el agente. Al considerar que la herramienta también puede ser útil para los expertos, la creación de perfiles separados les permitiría acceder a la herramienta y proporcionar un espacio que les ayude a dar explicaciones sobre las recomendaciones realizadas por el agente.

Un último aspecto a tener en cuenta es la integración de las interfaces de alimentos en la aplicación. Estas interfaces se han incluido con el propósito de permitir a los usuarios registrar su dieta diaria. Esta funcionalidad no solo ayudaría a obtener información más detallada de los usuarios, sino que les permitiría mantener un historial de su alimentación. Al contar con un registro completo de su dieta, los usuarios podrían tener una visión más precisa de su alimentación y sus posibles interacciones con ella cuando se sientan mal o experimenten algún malestar.

La integración de las interfaces de alimentos permitiría a la aplicación visualizar de manera integral toda la información relacionada con la dieta de los usuarios. Esto incluiría datos más profundos y detallados, lo que a su vez facilita el análisis de las posibles interacciones entre los alimentos consumidos. Al tener acceso a un mayor conjunto de datos sobre la alimentación de los usuarios, la aplicación podría brindar recomendaciones más precisas y personalizadas, así como ofrecer información relevante para una toma de decisiones sobre la dieta.

La integración de esta funcionalidad en la aplicación tendría un impacto significativo al proporcionar una visión más completa y detallada de la alimentación de los usuarios, lo cual desempeñaría un papel fundamental en la promoción de hábitos alimentarios saludables y la prevención de posibles interacciones entre medicamentos y alimentos. Además, esta mejora contribuiría a mejorar la experiencia general del usuario al brindar información más precisa, contextualizada y adaptada a sus necesidades y preferencias. Por lo tanto, los trabajos futuros representan una oportunidad valiosa para potenciar la aplicabilidad y utilidad de esta herramienta, generando un impacto positivo en la salud y el bienestar de los usuarios.

Referencias

- Allouch, M., Azaria, A., & Azoulay, R. (2021). Conversational Agents: Goals, Technologies, Vision and Challenges. *Sensors*, 21(24), 8448. <https://doi.org/10.3390/s21248448>
- Ameur, R. & Heudin, J. C. (2006). Interactive Intelligent Agent Architecture. 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Workshops. <https://doi.org/10.1109/wi-iatw.2006.83>
- Arora, S., Batra, K., & Singh, S. (2013). Dialogue system: A brief review. arXiv preprint arXiv:1306.4134.
- Arrabales, R. (2020). Perla: A Conversational Agent for Depression Screening in Digital Ecosystems. Design, Implementation and Validation. arXiv.org. <https://arxiv.org/abs/2008.12875>
- Artificial Solutions (2020). Chatbots. <https://www.artificialsolutions.com/chatbots>.
- Augustin, L. S. A., Kendall, C. W. C., Jenkins, D. J. A., Willett, W. C., Astrup, A., Barclay, A. W., Björck, I., Brand-Miller, J. C., Brighenti, F., Buyken, A. E., Ceriello, A., La Vecchia, C., Livesey, G., Liu, S., Riccardi, G., Rizkalla, S. W., Sievenpiper, J. L., Trichopoulou, A., Wolever, T. M. S., Baer-Sinnott, S., ... Poli, A. (2015). Glycemic index, glycemic load and glycemic response: An International Scientific Consensus Summit from the International Carbohydrate Quality Consortium (ICQC). *Nutrition, metabolism, and cardiovascular diseases: NMCD*, 25(9), 795–815. <https://doi.org/10.1016/j.numecd.2015.05.005>
- Azevedo, R. F. L., Morrow, D. G., Graumlich, J. F., Willemsen-Dunlap, A., Hasegawa-Johnson, M., Huang, T. S., Gu, K., Bhat, S., Sakakini, T., Sadauskas, V. & Halpin, D. J. (2017). Using conversational agents to explain medication instructions to older adults. American Medical Informatics Association Annual Symposium, 2018, 185-194.
- Bressan Martin, G. S. (2020). Interacción Fármaco-Nutriente.
- Chanampe, H., Aciar, S., Vega, M. D. L., Molinari Sotomayor, J. L., Carrascosa, G., & Lorefice, A. (2019). Modelo de redes neuronales convolucionales profundas para la clasificación de lesiones en ecografías mamarias. In XXI Workshop de Investigadores en Ciencias de la Computación (WICC 2019, Universidad Nacional de San Juan)..
- Cuevas, J. J., Martínez, E. P., Cortés, J. D. J. G., Pérez, S. S., & Campos, J. A. C. (2020). COMPARATIVA DE DESEMPEÑO DE LOS OPTIMIZADORES ADAM VS SGD EN EL ENTRENAMIENTO DE REDES NEURONALES CONVOLUCIONALES PARA LA CLASIFICACIÓN DE IMÁGENES ECG (COMPARATIVE PERFORMANCE OF ADAM VS. SGD OPTIMIZERS IN CONVOLUTIONAL NEURAL NETWORK TRAINING FOR THE CLASSIFICATION OF ECG IMAGES). *Pistas Educativas*, 42(137)

- Díaz SM, X., Neri, D., Moraga, F., Rebollo, M. J., Olivares, S., & Castillo, C. (2006). Análisis comparativo de la canasta básica de alimentos, pirámide alimentaria y recomendaciones nutricionales para preescolares y escolares chilenos: A comparative analysis. *Revista chilena de pediatría*, 77(5), 466-472.
- Dingler, T., Kwasnicka, D., Wei, J., Gong, E. & Oldenburg, B. (2021b). The Use and Promise of Conversational Agents in Digital Health. *Yearbook of Medical Informatics*, 30(01), 191-199. <https://doi.org/10.1055/s-0041-1726510>
- Drier, S. (2020). Avoiding food-drug interactions: A proposal. *Journal of family medicine and disease prevention*, 6(3). <https://doi.org/10.23937/2469-5793/1510128>
- Eleazu C. O. (2016). The concept of low glycemic index and glycemic load foods as panacea for type 2 diabetes mellitus; prospects, challenges and solutions. *African health sciences*, 16(2), 468–479. <https://doi.org/10.4314/ahs.v16i2.15>
- Firestore. (s.f.). Documentación de Firestore. <https://firebase.google.com/docs?hl=es-419>
- Flutter_tts. (2018). Dart Packages. https://pub.dev/packages/flutter_tts
- Fortes, P. (2020). Envejecimiento y atención a la dependencia en Ecuador. Catalogación en la fuente proporcionada por la Biblioteca Felipe Herrera del Banco Interamericano de Desarrollo. <https://publications.iadb.org/publications/spanish/document/Envejecimiento-y-atencion-a-la-dependencia-en-Ecuador.pdf>
- García, R. (2021). El perceptrón: una red neuronal artificial para clasificar datos. *Revista de investigación en modelos matemáticos aplicados a la gestión de la economía*.
- Gil, A. (2019). Muestreo. <https://www.studocu.com/ca-es/document/universitat-oberta-de-catalunya/fundamentos-de-estadistica/modulo-2-muestreo-apuntes/15124392>
- Goel, R., Vashisht, S., Dhanda, A., & Susan, S. (2021). An Empathetic Conversational Agent with Attentional Mechanism. <https://doi.org/10.1109/iccci50826.2021.9402337>
- Google Cloud. (s.f.). Dialogflow Documentation. <https://cloud.google.com/dialogflow/docs?hl=es-419>
- György, M., & Zoltán, S. (2018). The Role of Chatbots in Formal Education. Subotica, Serbia. IEEE. <https://ieeexplore.ieee.org/document/8524609>.
- Hao, J., & Ho, T. K. (2019). Machine learning made easy: a review of scikit-learn package in python programming language. *Journal of Educational and Behavioral Statistics*, 44(3), 348-361.
- Harvard Medical School. (2021). The lowdown on glycemic index and glycemic load. Harvard Health Publishing. <https://www.health.harvard.edu/diseases-and-conditions/the-lowdown-on-glycemic-index-and-glycemic-load>

- Holmes, S., Moorhead, A., Bond, R., Zheng, H., Coates, V., & McTear, M. F. (2019). Usability testing of a healthcare chatbot: Can we use conventional methods to assess conversational user interfaces? <https://doi.org/10.1145/3335082.3335094>
- IDF Diabetes Atlas. (2021). [Diabetesatlas.org](https://diabetesatlas.org/). <https://diabetesatlas.org/>
- IBM. (2023). WebSocket. <https://www.ibm.com/docs/en/was-liberty/core?topic=liberty-websocket>
- Instituto Nacional de Estadística y Censos. (2017). Diabetes, segunda causa de muerte después de las enfermedades isquémicas del corazón. Instituto Nacional de Estadística y Censos. <https://www.ecuadorencifras.gob.ec/diabetes-segunda-causa-de-muerte-despues-de-las-enfermedades-isquemicas-del-corazon/>
- Izaurieta, F., & Saavedra, C. (2000). Redes neuronales artificiales. Departamento de Física, Universidad de Concepción Chile.
- Jia, Z., Yang, Y., Gao, W. & Chen, X. (2015b). User-Based Collaborative Filtering for Tourist Attraction Recommendations. 2015 IEEE International Conference on Computational Intelligence & Communication Technology. <https://doi.org/10.1109/cict.2015.20>
- Keays, R. (2019). Diabetes. *Current Anaesthesia and Critical Care*, 18(2), 69–75. <https://doi.org/10.1016/j.cacc.2007.03.007>
- Kim, D. (2020). Glycemic index. In *Obesity: Global Impact and Epidemiology* (pp. 183–189). Elsevier. <https://doi.org/10.1016/B978-0-12-818839-2.00014-4>
- Lo, Y. M. O., & Abihm, M. D. (2018). Glycemic index and glycemic load. In D. P. Rakel (Ed.), *Integrative medicine* (4th ed., p. 1). Elsevier.
- López, D. (2014). Fármaco-nutrimiento: interacción insuficientemente considerada. *Revista de Especialidades Médico-Quirúrgicas*, 19(2), 244-250.
- López, M. (2021). Servicios cognitivos para el diseño de agentes conversacionales: Un enfoque teórico-práctico. Universidad De Málaga. <https://riuma.uma.es/xmlui/handle/10630/23515>
- Lu, Z. Q., & Yan, J. (2014). Dietary Carbohydrate and Age-Related Cataract. In *Handbook of Nutrition, Diet and the Eye* (pp. 271–277). Elsevier Inc. <https://doi.org/10.1016/B978-0-12-401717-7.00027-7>
- Martin, J., Muñoz, C., & Ábalos, N. (2017). Improve your chatbot's design. <https://chatbottest.com/#learn-more>
- Mehrotra, A. & Musolesi, M. (2017). Intelligent Notification Systems: A Survey of the State of the Art and Research Challenges. *arXiv: Human-Computer Interaction*.
- Meneses, K. (2020). Nueva Guía: Tabla de alimentos, Índice Glucémico y Carga Glucémica. Centro Médico D-Médical; D-Medical. <https://d-medical.com/2020/06/nueva-tabla-de-alimentos-indice-glucemico-y-carga-glucemica/>

- Microsoft. (s.f.). IA conversacional | Microsoft Power Virtual Agents. <https://powervirtualagents.microsoft.com/es-es/conversational-ai/>
- Mozilla Developer Network. (2023). Map - JavaScript. https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Map
- Mugoye, K., Okoyo, H., & Mcoyowo, S. (2019). Smart-bot Technology: Conversational Agents Role in Maternal Healthcare Support. <https://doi.org/10.23919/istafrica.2019.8764817>
- Navarrete, P. J., Velasco, J. C., & Loro, L. M. (2021). Automedicación en época de pandemia: Covid-19. *Revista del cuerpo médico del HNAAA*, 13(4), 350-355. <https://doi.org/10.35434/rcmhnaaa.2020.134.762>
- Nuruzzaman, M., & Hussain, O. K. (2018). A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks. <https://doi.org/10.1109/icebe.2018.00019>
- Olguín, G. E. M., & De Jesús, Y. L. (2019). Métricas de similaridad y evaluación para sistemas de recomendación de filtrado colaborativo. *Revista de Investigación en Tecnologías de la Información: RITI*, 7(14), 224-240.
- Partida, V. (2022). Los determinantes demográficos del envejecimiento de la población. <https://rppoblacion.uaemex.mx/article/view/18178>
- Preininger, A. M., South, B., Heiland, J., Buchold, A., Baca, M., Wang, S., Nipper, R., Kutub, N., Bohanan, B. & Jackson, G. P. (2020). Artificial intelligence-based conversational agent to support medication prescribing. *JAMIA Open*, 3(2), 225-232. <https://doi.org/10.1093/jamiaopen/ooaa009>
- RedHat. (2023). What is a webhook?. <https://www.redhat.com/en/topics/automation/what-is-a-webhook>
- Ryan, J. L., Mazzuchi, T. A., Ryan, D. H., De La Cruz, J. L., & Cooke, R. M. (2012). Quantifying information security risks using expert judgment elicitation. *Computers & Operations Research*, 39(4), 774–784. <https://doi.org/10.1016/j.cor.2010.11.013>
- Sánchez, C., Sevillano, J., & Fernández, L. (2017). Healthrecsys: Sistema recomendador para la salud [Tesis Doctoral, Universidad de Sevilla]. https://idus.us.es/bitstream/handle/11441/76323/HealthRecSys_Sistema%20recomendador%20en%20Salud_DEF.pdf?sequence=1&isAllowed=y
- Singh, E., Bompelli, A., Wan, R., Bian, J., Pakhomov, S. V. S., & Zhang, R. (2022). A conversational agent system for dietary supplements use. *BMC Medical Informatics and Decision Making*, 22(S1). <https://doi.org/10.1186/s12911-022-01888-5>
- Singh, R. K., Paste, M., Shinde, N. V., Patel, H., & Mishra, N. (2018). Chatbot using TensorFlow for small Businesses. <https://doi.org/10.1109/icicct.2018.8472998>
- Speech to text. (2019, October 4). Dart Packages. https://pub.dev/packages/speech_to_text
- Supabase. (s.f.). Supabase Documentation. <https://supabase.com/docs>

- TensorFlow, (2023). API Documentation. https://www.tensorflow.org/api_docs
- The University of Sydney. (2023). Glycemic index – glycemic index research and GI news. Glycemicindex.com. <https://glycemicindex.com/>
- Uman, I. (2018). El efecto Netflix: cómo los sistemas de recomendación transforman las prácticas de consumo cultural y la industria de contenidos. Cuadernos de comunicólogos, 6, 27-42.
- Villalta, S. J. P., Tulcanaza, C. A. M., Carvajal, J. P. E., Abril, C., Hurtado, S., López, C. B., & Buse, J. R. (2018). Automedicación en la región interandina norte del Ecuador: una práctica usual. Revista de la Facultad de Ciencias Médicas (Quito), 43(2), 78-85. <https://doi.org/10.29166/rfcmq.v43i2.2824>
- Vo, T. H., Nguyen, N. T. K., Kha, Q. H., & Le, N. Q. K. (2022). On the road to explainable AI in drug-drug interactions prediction: A systematic review. Computational and Structural Biotechnology Journal.
- Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., Assempour, N., Iynkkaran, I., Liu, Y., Maciejewski, A., Gale, N., Wilson, A., Chin, L., Cummings, R., Le, D., Pon, A., ... Wilson, M. (2018). DrugBank 5.0: a major update to the DrugBank database for 2018. Nucleic acids research, 46(D1), D1074–D1082. <https://doi.org/10.1093/nar/gkx1037>
- Xiong, G., Yang, Z., Yi, J., Wang, N., Wang, L., Zhu, H., Wu, C., Lu, A., Chen, X., Liu, S., Hou, T., & Cao, D. (2022). DDInter: An online drug-drug interaction database towards improving clinical decision-making and patient safety. Nucleic Acids Research, 50(D1), D1200–D1207. <https://doi.org/10.1093/nar/gkab880>
- Zabala, A., & Fernández, E. (2018). Diabetes mellitus tipo 2 en el Ecuador: revisión epidemiológica. *Medicinas UTA*, 2(4), 3–9. <https://revistas.uta.edu.ec/revista/index.php/medi/article/view/1219>

Anexos

Anexo A

Definición de escenario 1 para la evaluación del juicio del experto.

Escenario 1

Caso	Fármaco-Fármaco y Fármaco-Alimento		
Interacción			
Edad	56	Sexo	Masculino
Historial Médico	Diabetes tipo 2, con complicaciones recientes de colesterol		
Medicación Actual	Metformina y Simvastatina		
Situación Actual	El paciente, de 56 años y sexo masculino, presenta diabetes tipo 2 hace 5 años y está bajo tratamiento con Metformina. Hace aproximadamente 15 días, se le agregó Simvastatina debido a niveles altos de colesterol. El paciente ha experimentado dolor en los músculos sin razón aparente después de consumir toronja. El dolor muscular persiste desde hace algunos días y presenta una intensidad de leve a moderada.		
Preguntas para el Experto	<p>¿Existe alguna interacción conocida entre la metformina y la simvastatina?</p> <hr/> <p>¿Se deben tomar precauciones adicionales al combinar la metformina y la simvastatina en este paciente con diabetes tipo 2?</p> <hr/> <p>¿Existen posibles efectos adversos si se toman estos dos medicamentos y adicionalmente se consume toronja?</p>		

Anexo B

Definición de escenario 2 para la evaluación del juicio del experto.

Escenario 2

Caso	Fármaco-Fármaco y Fármaco-Alimento		
Interacción			
Edad	25	Sexo	Masculino
Historial Médico	Diabetes tipo 1 desde hace ocho años, y ahora presenta hipertensión		
Medicación Actual	Insulina NPH 22 UI antes del desayuno y 8 UI antes de la merienda y Enalapril 20 mg con el desayuno		
Situación Actual	El paciente, de 25 años de edad y sexo masculino, padece diabetes tipo 1 y requiere administrarse insulina. Además, presenta hipertensión y está tomando Enalapril. El paciente es habitual de consumir uno de los siguientes alimentos: plátano, chirimoya, y coco fresco. Desea conocer si existen posibles interacciones entre los medicamentos y alguno de estos alimentos, y si se deben tomar precauciones adicionales		
Preguntas para el Experto	<p>¿Se deben tener precauciones adicionales al combinar la Insulina y el Enalapril para controlar la diabetes en este paciente con diabetes tipo 1 e hipertensión?</p> <p>¿El Enalapril puede tener alguna interacción conocida con alguno de los alimentos mencionados (plátano, chirimoya, y coco fresco)?</p> <p>Considerando las necesidades dietéticas y la combinación de medicamentos del paciente, ¿Qué alternativas de consumo de alimentos se podrían ofrecer además de los mencionados anteriormente para evitar posibles interacciones?</p>		

Anexo C

Definición de escenario 3 para la evaluación del juicio del experto.

Escenario 3

Caso	Fármaco-Alimento		
Interacción			
Edad	64	Sexo	Femenino
Historial Médico	Diabetes tipo 2 desde hace 13 años.		
Medicación Actual	Metformina 1000 mg y Sitagliptina 50 mg para el manejo de la diabetes tipo 2.		
Situación Actual	La paciente, de 64 años y sexo femenino, está tomando Metformina y Sitagliptina para el control de la diabetes tipo 2. Desea consumir una manzana y además planea beber una cerveza. La mujer planea comer 3 manzanas y tiene además antojo de un par de cervezas personales para combatir el calor.		
Preguntas para el Experto	<p>¿Existe alguna interacción conocida entre los medicamentos que el paciente está tomando (metformina y sitagliptina) y la manzana que pueda afectar su salud o la efectividad de la medicación?</p> <hr/> <p>Con respecto a la cerveza, ¿Existe algún impedimento para que el paciente la consuma mientras esté tomando metformina y sitagliptina?</p> <hr/> <p>¿Podría la ingesta de alcohol afectar la eficacia de los medicamentos (metformina y sitagliptina) o causar algún efecto secundario en este paciente con diabetes tipo 2?</p>		

Anexo D

Definición de escenario 4 para la evaluación del juicio del experto.

Escenario 4

Caso	Fármaco-Fármaco		
Interacción			
Edad	48	Sexo	Femenino
Historial Médico	Diabetes tipo 2 desde hace 8 años.		
Medicación Actual	Glipizida 5 mg. para el manejo de la diabetes tipo 2.		
Situación Actual	La mujer de 48 años y sexo femenino, tiene diabetes tipo 2 y está tomando glipizida para su control. Recientemente, ha experimentado un dolor de cabeza intenso y ha decidido tomar ibuprofeno, un medicamento de venta libre comúnmente utilizado para aliviar el dolor de cabeza. El paciente tomó por dos ocasiones ibuprofeno de 600 mg para tratar el dolor.		
Preguntas para el Experto	Teniendo en cuenta la medicación actual del paciente (glipizida) para su diabetes tipo 2 y su decisión de tomar ibuprofeno para el dolor de cabeza, ¿Qué posibles interacciones medicamentosas podrían ocurrir y cuáles son las precauciones recomendadas?		