



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL
CARRERA DE TEGNOLOGÍAS DE LA INFORMACIÓN

Tema:

ASISTENTE MÉDICO VIRTUAL ORIENTADO AL DIAGNÓSTICO DE
ENFERMEDADES RESPIRATORIAS APLICANDO INTELIGENCIA
ARTIFICIAL

Trabajo de Integración Curricular Modalidad: Proyecto de investigación, presentado previo a la obtención del título de Ingeniero en Tecnologías de la Información.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: David Sebastian González Albán

TUTOR: Ing. Dennis Vinicio Chicaiza Castillo, Mg.

Ambato – Ecuador

marzo - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Integración Curricular con el tema: ASISTENTE MEDICO VIRTUAL ORIENTADO AL DIAGNÓSTICO DE ENFERMEDADES RESPIRATORIAS APLICANDO INTELIGENCIA ARTIFICIAL, desarrollado bajo la modalidad Proyecto de Investigación por el señor David Sebastian González Albán, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 de las segundas reformas al Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y el numeral 7.4 del respectivo instructivo del reglamento.

Ambato, marzo 2023.

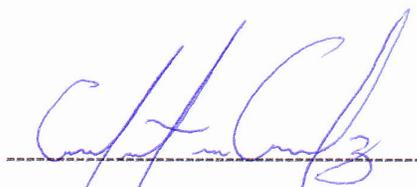
Ing. Dennis Vinicio Chicaiza Castillo

TUTOR

AUTORÍA

El presente trabajo de Integración Curricular titulado: ASISTENTE MÉDICO VIRTUAL ORIENTADO AL DIAGNÓSTICO DE ENFERMEDADES RESPIRATORIAS APLICANDO INTELIGENCIA ARTIFICIAL es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2023.



David Sebastian González Albán

C.C. 1804776696

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Integración Curricular como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Integración Curricular en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, marzo 2023.



Handwritten signature in blue ink, appearing to read 'David Sebastian González Albán', positioned above a horizontal dashed line.

David Sebastian González Albán

C.C. 1804776696

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Integración Curricular presentado por el señor David Sebastian González Albán, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado ASISTENTE MEDICO VIRTUAL ORIENTADO AL DIAGNÓSTICO DE ENFERMEDADES RESPIRATORIAS APLICANDO INTELIGENCIA ARTIFICIAL, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 de las segundas reformas al Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y al numeral 7.6 del respectivo instructivo del reglamento. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, marzo 2023.

Ing. Elsa Pilar Urrutia Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Franklin Oswaldo Mayorga Mayorga

PROFESOR CALIFICADOR

Ing. Rubén Eduardo Nogales Portero

PROFESOR CALIFICADOR

DEDICATORIA

El presente trabajo de investigación es dedicado a mi abuelito que lamentablemente no pudo estar para presenciar este momento importante en mi vida, a mi abuelita que ha sido una maestra y ha sabido inculcarme el orden y la disciplina, a mi madre y hermanas por el apoyo que me han brindado.

A mi primo que ha cumplido el papel de hermano, mejor amigo y de ser el guía y apoyo que he necesitado para poder ser la persona que hoy soy, siempre le estaré agradecido por todo el cariño que me ha brindado.

A mi mejor amigo que ha sido una persona excepcional, y le estaré eternamente agradecido por el apoyo que me ha brindado en el ámbito educativo y en si como un amigo, motivándome siempre que tome buenas decisiones y vea lo mejor por mi futuro.

David Sebastian González Albán

AGRADECIMIENTO

Agradezco a mis profesores y compañeros que he conocido en el proceso de realizar mi carrera por el apoyo que me han brindado para poder llegar lejos.

A mi familia que me ha sabido ayudar en momentos que tenía problemas en el ámbito educativo.

A mis amigos que me han apoyado guiándome para poder cumplir con las metas propuestas.

A una persona especial que me supo brindar el apoyo en un momento crítico de mi vida y me supo dar la mano de manera desinteresada y gracias a ello he crecido como persona y profesionalmente.

Agradezco de manera especial a mi tutor por guiarme durante el desarrollo del proyecto de investigación.

ÍNDICE DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE DE CONTENIDOS	viii
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE TABLAS	xiii
RESUMEN EJECUTIVO.....	xiv
ABSTRACT	xv
CAPITULO 1.- MARCO TEÓRICO.....	1
1.1. Tema de Investigación	1
1.1.1. Planteamiento del Problema	1
1.2. Antecedentes Investigativos.....	3
1.3. Fundamentación Teórica.....	5
1.3.1. Enfermedades respiratorias.....	5
1.3.2. Tipos de enfermedades respiratorias.....	6
1.3.3. Enfermedades de vías respiratorias superior	8
1.3.4. Desarrollo Web	8
1.3.5. Aplicación móvil.....	9
1.3.6. Tipos de aplicaciones móviles	9
1.3.7. Inteligencia Artificial.....	10

1.3.8.	Tipos de Inteligencia Artificial	11
1.3.9.	Flutter	11
1.3.10.	Dart	12
1.3.11.	Firebase	13
1.3.12.	Railway	13
1.3.13.	Metodologías Ágiles	13
1.3.14.	Metodologías Ágiles para el desarrollo de aplicaciones móviles	14
1.4.	Objetivos	15
1.4.1.	Objetivo General	15
1.4.2.	Objetivos Específicos	15
CAPITULO II.- METODOLOGÍA		16
2.1.	Materiales	16
Estructura del Dataset		17
2.2.	Métodos	25
2.2.1.	Modalidad de Investigación	25
	Investigación Documental	25
	Investigación Experimental	25
	Investigación Aplicada	25
	Investigación Bibliográfica-documental	26
2.2.2.	Población y Muestra	26
2.2.3.	Recolección de Información	27
2.2.4.	Procesamiento y análisis de datos	27
CAPITULO III.- RESULTADOS Y DISCUSIÓN		29
3.1.	Análisis y Discusión de los Resultados	29
3.1.1.	Algoritmos para el desarrollo de asistentes virtuales	29

3.1.2. Comparación entre algoritmos de asistentes virtuales.....	32
3.1.3. Selección del Algoritmo para el desarrollo de asistentes virtuales	33
3.1.4. Frameworks para el desarrollo de aplicaciones móviles	33
3.1.5. Tabla Comparativa de Frameworks para aplicaciones móviles	34
3.1.6. Selección del framework para el desarrollo de la aplicación móvil.....	35
3.1.7. Metodología.....	36
3.2. Metodología Mobile -D	37
3.2.1. Fase de Exploración	37
3.2.2. Fase de Inicialización	38
3.2.3. Fase de Estabilización	66
3.2.4. Fase de Pruebas	67
CAPITULO IV.- CONCLUSIONES Y RECOMENDACIONES	72
4.1. Conclusiones	72
4.2. Recomendaciones.....	73
BIBLIOGRAFÍA	74
ANEXOS	78

ÍNDICE DE FIGURAS

Figura 1. Sintomatología de las enfermedades respiratorias	17
Figura 2. celdas vacías y celdas llenas.....	18
Figura 3. Incidencia de la sintomatología en cada enfermedad respiratoria.....	20
Figura 4. Comparativa entre dolor de garanta y género.....	21
Figura 5. Comparativa Dolor Articular y Género	22
Figura 6. Comparativa entre dolor de cabeza y género.....	23
Figura 7. Comparativa entre género y tos.....	24
Figura 8. Árbol de Decisiones.....	29
Figura 9. Random Forest.....	30
Figura 10. Support Vector Machine.....	31
Figura 11. Arquitectura del proyecto	39
Figura 12. Colección Base de Datos	40
Figura 13. Documentos Base de Datos	41
Figura 14. Ejemplo Base de Datos.....	42
Figura 15. Pantalla de Inicio del asistente médico virtual	44
Figura 16. Pantalla de Primera Pregunta.....	45
Figura 17. Pantalla de la quinta pregunta.....	46
Figura 18. Pantalla numero 15 del app	47
Figura 19. Pregunta numero 20 del app	48
Figura 20. Pantalla de última pregunta	49
Figura 21. Pantalla de Diagnóstico	50
Figura 22. Función Predecir Enfermedad	51
Figura 23. Obtener temperatura del sensor	52
Figura 24. Conexión a la API en el host	52
Figura 25. Preguntas en Flutter	53
Figura 26. Inicializar los valores al aperturar el app.....	54
Figura 27. Definición de clases usadas en el dataset	56
Figura 28. Función Transformar los datos	57

Figura 29. Separación de datos	57
Figura 30. Balanceo y división de entrenamiento.....	58
Figura 31. Entrenamiento del modelo.....	58
Figura 32. Creación de la Matriz de Confusión	58
Figura 33. Procesamiento de la Matriz de Confusión	59
Figura 34. Validación Cruzada	60
Figura 35. Resultado de la Validación Cruzada.....	60
Figura 36. Guardar el modelo en un archivo.pkl	60
Figura 37. Carga del modelo Random Forest y predicción	61
Figura 38. Creación de la api	62
Figura 39. Creación de Proyecto Railway	63
Figura 40. Estructura proyecto FastAPI - Railway	64
Figura 41. Librerías y sus versiones en formato Railway	64
Figura 42. Cadena de conexión a Firebase	65
Figura 43. Función guardar registro.....	65
Figura 44. Función obtener temperatura	66
Figura 45. API ejecutándose en Railway	66
Figura 46. Pantalla de Bienvenida	68
Figura 47. Obtención de la temperatura desde el dispositivo	69
Figura 48. Mensaje de diagnostico en la pantalla final.....	70

ÍNDICE DE TABLAS

Tabla 1. Población de estudio	27
Tabla 2. Segmentación de sintomatología de las enfermedades respiratorias.	28
Tabla 3. Tabla comparativa de algoritmos para asistentes virtuales	32
Tabla 4. Tabla comparativa de frameworks para aplicaciones móviles.....	34
Tabla 5. Tabla comparativa de metodologías ágiles	36
Tabla 6. Descripción del dataset	55
Tabla 7. Prueba de aceptación.....	71

RESUMEN EJECUTIVO

Actualmente, las enfermedades respiratorias afectan de manera significativa a una gran parte de la población a nivel mundial, alrededor de 3,9 millones de personas mueren anualmente por infecciones respiratorias agudas; entre estas, las enfermedades pulmonares son la principal causa de muerte, y sus causas pueden estar influenciadas por el tabaquismo y la exposición a la contaminación ambiental.

Este proyecto, tiene como objetivo crear una aplicación que permita diagnosticar de manera más ágil las enfermedades respiratorias e incentivar a la población a buscar atención médica frente a estas afecciones. Para ello se buscó crear un asistente médico virtual; primero, se utilizó el método de observación como herramienta de análisis; segundo, se usaron como referencia datos obtenidos del Hospital San Cayetano, mismos que se correlacionaron con información recopilada por una pulsera electrónica que nos arroja resultados de distintas variables (pulso, temperatura y saturación), esta pulsera es parte del proyecto de investigación “Sistema Electrónico de Monitoreo de Bioseñales y Alertas de Distanciamiento Social para la Prevención del SARS-COV2 aplicando Inteligencia Artificial”; en tercer lugar, se usó toda esta información para entrenar a la IA permitiendo la creación de una aplicación efectiva en el diagnóstico de enfermedades respiratorias.

La aplicación móvil se creó con base en la metodología Mobile - D, logrando un desarrollo ágil del proyecto; se usó el framework de Flutter para el desarrollo con lenguaje de programación Dart, se alojó el rest-api en el servicio de Railway, ya que esto permitió una mayor facilidad para alojar la api dedicada. Finalmente, la base de datos fue desarrollada en Firebase de Google, ya que ofrece una mayor facilidad de implementación al momento de generar un llamado desde el aplicativo hacia la base de datos.

Palabras clave: Asistente Médico Virtual, Tecnologías de la Información, aplicación móvil, enfermedades respiratorias, metodología Mobile – D, Flutter.

ABSTRACT

Currently, respiratory diseases significantly affect a large part of the population worldwide, around 3.9 million people die annually from acute respiratory infections; among these, lung diseases are the leading cause of death, and their causes may be influenced by smoking and exposure to environmental pollution.

The objective of this project is to create an application that allows a more agile diagnosis of respiratory diseases and encourages the population to seek medical attention for these conditions. For this, we sought to create a virtual medical assistant; first, the observation method was used as an analysis tool; second, data obtained from the San Cayetano Hospital were used as a reference, which were correlated with information collected by an electronic bracelet that gives us results of different variables (pulse, temperature and saturation), this bracelet is part of the research project "Electronic System Monitoring of Biosignals and Social Distancing Alerts for the Prevention of SARS-COV2 applying Artificial Intelligence"; thirdly, all this information was used to train the AI allowing the creation of an effective application in the diagnosis of respiratory diseases.

The mobile application was created based on the Mobile - D methodology, achieving an agile development of the project; The Flutter framework was used for development with the Dart programming language, the rest-api was hosted in the Railway service, since this allowed it to be easier to host the dedicated api. Finally, the database was developed in Google's Firebase, since it offers greater ease of implementation when generating a call from the application to the database.

Keywords: Virtual Medical Assistant, information technologies, mobile application, respiratory diseases, Mobile – D methodology, Flutter.

CAPITULO 1.- MARCO TEÓRICO

1.1. Tema de Investigación

ASISTENTE MÉDICO VIRTUAL ORIENTADO AL DIAGNÓSTICO DE ENFERMEDADES RESPIRATORIAS APLICANDO INTELIGENCIA ARTIFICIAL

1.1.1. Planteamiento del Problema

Las enfermedades respiratorias implican una gran carga para los sistemas de salud, por lo que entre las 10 principales causas de muerte en todo el mundo reportadas por la Organización Mundial de la Salud (OMS) se encuentran las infecciones del tracto respiratorio inferior. Dentro de este grupo de enfermedades podemos mencionar a la Enfermedad Pulmonar Obstructiva Crónica (EPOC), misma que ha afectado a alrededor de 200 millones de personas, la Tuberculosis (TB) que ha llegado a afectar a 8,7 millones de personas y el cáncer de pulmón como la enfermedad cancerígena más frecuente en el mundo que mata a más de 1,4 millones de personas cada año. La EPOC y el cáncer de pulmón muestran una morbilidad y mortalidad crecientes representando el 16,7% de las muertes, frente a la enfermedad coronaria con el 12,8% de muertes en el mundo. La EPOC, es la cuarta causa principal de muerte en el mundo y su desarrollo muchas veces se ve influido por el consumo de tabaco, ya que provoca la destrucción del tejido pulmonar y la obstrucción de las vías respiratorias pequeñas, dando lugar a la bronquitis y el enfisema que son las principales enfermedades de la EPOC; por otro lado, el cáncer de pulmón es la causa más frecuente de mortalidad por cáncer, en esta enfermedad también tiene gran influencia el humo del tabaco, sin embargo, los efectos se relacionan directamente con la cantidad y la duración del tabaquismo, ya que el cáncer se desarrolla años después de que las personas empiecen a fumar. Dicho esto, se puede argumentar que, la prevención, curación y control de las enfermedades respiratorias deben ser las principales prioridades en la toma de decisiones globales en el sector sanitario; además de ello, promover una mayor inversión en salud respiratoria proporcionará diversos frutos en cuanto a longevidad, días de vida sana y economías nacionales [1].

La OMS estima que alrededor de 3,9 millones de personas mueren anualmente por infecciones respiratorias agudas. Solo la influenza estacional puede alcanzar alrededor de 600 millones de casos anuales en todo el mundo, de los cuales 3 millones son graves, con una tasa de mortalidad estimada de 250 000 a 500 000. Los lactantes y los niños más pequeños son especialmente propensos a padecer estas enfermedades, nueve millones de niños menores de 5 años mueren cada año y las enfermedades respiratorias son la causa más frecuente de estas muertes; entre ellas, la neumonía es la principal causa de mortalidad del mundo en los niños pequeños y el asma es la enfermedad crónica más frecuente, que afecta a alrededor del 14 % de los niños en todo el mundo [1].

América Latina está formada por diversos países que se encuentran en vías de desarrollo, posee alrededor de 600 millones de habitantes lo que representa al 8,2% de la población mundial. Esta región del mundo plantea grandes retos para la salud general y respiratoria. La mayor parte de los países muestran tasas significativas de enfermedades respiratorias crónicas o de exposiciones de riesgo; además de ello, la disponibilidad de recursos humanos para la salud es escasa, sobre todo si nos referimos a especialistas en enfermedades respiratorias; y los centros de formación académica son pocos e incluso inexistentes en la mayor parte de los países [2].

En la actualidad, gracias al desarrollo de la tecnología y a las facilidades que ahora nos brinda, existe la posibilidad de recibir cierto tipo de atención médica, este es el caso de la empresa Justanswer ubicada en Estados Unidos, ellos han logrado desarrollar un asistente multiprofesional virtual para solventar los requerimientos de todos los usuarios que se benefician de sus servicios, logrando brindar un servicio garantizado y de calidad, atención personalizada e ininterrumpida por su atención 24/7 y un ahorro de tiempo y dinero al no generar gastos por movilización, de esta manera se logró dar espacios y tiempos óptimos de atención a los usuarios evitando el colapso de los centros de atención médica y proporcionando un método más ágil para precautelar la salud de las personas haciendo uso de herramientas tecnológicas [3].

En Ecuador no existe un asistente médico virtual implementado, sin embargo se ha creado un proyecto en estado de prototipo denominado Pillbot, mismo que busca implementar un sistema inteligente para prevenir y dar tratamiento a pacientes con enfermedades crónicas brindando atención médica, seguimiento y educación en salud [4]. Es por esta razón que la Universidad Técnica de Ambato con el fin de obtener nuevo conocimiento para el beneficio de la comunidad, busca crear un asistente médico virtual orientado al diagnóstico de enfermedades respiratorias específicamente: Rinitis Alérgica, Gripe Común y Covid19; este trabajo de investigación se integra al proyecto macro denominado Sistema Electrónico de Monitoreo de Bioseñales y Alertas de Distanciamiento Social para la Prevención de SARS-COV2 aplicando Inteligencia Artificial.

1.2. Antecedentes Investigativos

Al realizar la revisión de la literatura pertinente, se encontraron trabajos de investigación en repositorios nacionales como también internacionales, los cuales aportan de manera significativa para el desarrollo del presente proyecto con temas e información similar al tema planteado.

Este es el caso de [5] quién al desarrollar un chatbot para brindar una mejor experiencia al cliente en un hotel de Colombia atendiendo diversas solicitudes acerca de alojamiento, tarifas, servicios, horarios y recomendaciones y enlazando este chatbot con los canales de comunicación del hotel, logró atender más del 75 % de las solicitudes generando un mecanismo de respuesta similar a como funciona: Siri, Alexa y Google Assistant, ya que este bot investiga inicialmente en su base de datos y en caso de no encontrar una respuesta acertada investiga en internet, de esta manera se logra cubrir el requerimiento presentado por el cliente ya que las solicitudes de información son atendidas por chatbots o asistentes virtuales que son más rápidos al momento de brindar información acerca de la página web del hotel.

También está [6] que en su proyecto para desarrollar un asistente virtual (chatbot) y facilitar el acceso a la información requerida por los estudiantes de la Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil, sobretodo en temas de

matriculación y titulación, determinó que el acceso a dicha información mejoró gracias a la ayuda del chatbot, obteniendo un porcentaje de aceptación de los estudiantes de un 75%; además de ello, se desarrolló un nuevo hábito a través de la aplicación de este chatbot, puesto que cada alumno utilizó la herramienta al menos una vez a pesar de que el método y medio de comunicación oficial es el correo electrónico y la página oficial.

De acuerdo con [7] al desarrollar un prototipo de aplicación móvil utilizando SDK FLUTTER y lenguaje de código abierto DART con el objetivo de incentivar a las personas a dejar el sedentarismo y ser parte de una comunidad físicamente activa incursionando en diversas áreas del deporte dependiendo de sus intereses, se determinó que existe una tendencia positiva hacia el uso de una aplicación que promueva la realización de actividades deportivas en jóvenes y adultos. En esta app se creó un módulo de eventos para que los usuarios puedan saber lo que se tiene planificado a realizar en la ciudad; y un módulo de geolocalización para que los usuarios puedan postear y visualizar el contenido.

Como menciona [8] en su estudio realizado a partir del análisis de las historias clínicas de una muestra de 588 pacientes con enfermedades respiratorias agudas (ERA) acerca de los factores de riesgo que implica tener una enfermedad ERA en menores de cinco años, logró identificar que los factores que mayor incidencia tienen para desarrollar Enfermedades Respiratorias Agudas son periodos de lactancia materna menor a seis meses, los antecedentes respiratorios, el grupo de edad, además identificó que las instituciones educativas a las que asisten estos niños no ofrecen condiciones apropiadas de salud para los niños, lo que influye directamente en que ellos tengan riesgo de enfermar de neumonía, rinofaringitis aguda catarral, bronquiolitis, otitis media aguda y faringoamigdalitis.

Existe también un proyecto realizado por [9] en donde se propone crear un sistema electrónico de monitoreo de signos vitales y alertas de distanciamiento social para prevenir el contagio y propagación de enfermedades virales respiratorias, aquí pudo identificar que la mayor parte de sistemas electrónicos implementados utilizan tecnologías de comunicación inalámbricas para transmitir los datos del paciente, el monitoreo y el almacenamiento de datos. Por otro lado, al contrastar los datos de ritmo cardiaco, saturación de oxígeno y temperatura corporal adquiridos a través del dispositivo frente a

los datos de instrumentos médicos, se determinó que el sistema electrónico presenta menor confiabilidad a nivel general, esto quiere decir que los valores de confiabilidad fueron de 0,24; 0,22 y 0,36 respectivamente en el dispositivo, a diferencia de los instrumentos médicos, en donde los valores fueron de 0,49; 0,42 y 0,6 manteniendo el orden mencionado.

Finalmente, se tiene a [10] quien al investigar las formas de prevención y control de infecciones respiratorias agudas con tendencia epidémica y pandémica durante la atención sanitaria pudo determinar que la práctica del aislamiento y las medidas de higiene resultaron eficaces para disminuir la transmisión de los virus respiratorios; además, las mascarillas médicas (mascarillas de tipo quirúrgico o con filtro de partículas N95) demostraron un alto desempeño en temas de prevención de contagio, sin embargo, los datos probatorios no indicaron una superioridad de las mascarillas filtrantes N95 sobre las mascarillas quirúrgicas sencillas. Por otro lado, se menciona también que el equipo de protección personal busca dar mayor seguridad al usuario, sin embargo, es posible que la disponibilidad de estos equipos sea limitada y que sea inevitable reutilizarlos, eso debe hacerse únicamente en condiciones seguras, evitando el uso innecesario de los equipos de protección personal.

1.3. Fundamentación Teórica

1.3.1. Enfermedades respiratorias

Las enfermedades respiratorias son una de las principales causas de atención médica en todo el mundo, tanto ambulatoria como hospitalaria, y es una de las principales causas de muerte. Estas enfermedades afectan a toda la población, principalmente a menores de 5 años, y mayores de 65 años. Gran parte de las consultas, hospitalizaciones y de las muertes son las enfermedades al sistema respiratorio de origen infeccioso, incluida la neumonía [11].

La gripe y la bronquiolitis son las enfermedades respiratorias más frecuentes y graves, sin dejar de lado otras enfermedades respiratorias infecciosas como otitis media, faringitis y la bronquitis forman parte del grupo de Infecciones Respiratorias Agudas (IRA),

representan la primera razón para consultas con los servicios médicos. Estas condiciones se pueden evitar con medidas higiénicas (lavado de manos, higiene del baño), aislamiento respiratorio y, sobre todo, tener la vacunación necesaria para evitar el contagio de (gripe, neumocócica, bacterias trivalentes) [11].

Las enfermedades respiratorias afectan a los conductos que transportan el oxígeno y otros gases dentro y fuera de los pulmones; usualmente, ocasionan un estrechamiento u obstrucción de las vías respiratorias y pueden dar lugar a enfermedades como el asma, enfermedad pulmonar obstructiva crónica (EPOC), bronquiolitis y bronquiectasia [12]. Estas enfermedades respiratorias se pueden producir por factores pulmonares, cardiovasculares y emocionales; su origen tiene relación directa con infecciones, consumo de tabaco y exposición a agentes contaminantes existentes en el aire (polvo y gases tóxicos). En definitiva, una enfermedad o trastorno respiratorio es cualquier condición que afecta directamente al sistema respiratorio y se presentan con síntomas como: tos, silbidos y presión en el pecho, secreciones nasales o flemas, dificultad para respirar, cansancio, rinitis, picazón de la nariz, estornudos, entre otros [13].

1.3.2. Tipos de enfermedades respiratorias

Existen muchos tipos de enfermedades respiratorias, que pueden ser agudas o crónicas, y pueden afectar a personas de todas las edades y condiciones de salud. Entre las enfermedades respiratorias más comunes están el resfriado común, la gripe, la bronquitis, la neumonía, el asma y la enfermedad pulmonar obstructiva crónica (EPOC) y la más nueva que es el COVID-19. Algunas enfermedades respiratorias pueden ser causadas por factores ambientales, mientras que otras pueden ser causadas por factores genéticos o de estilo de vida [14]:

Gripe Común

Es una infección viral repentina, leve y generalmente autolimitada del tracto respiratorio superior que se desarrolla con inflamación de la mucosa nasofaríngea, interfiriendo de esta manera con el drenaje normal de la mucosidad, lo

cual provoca congestión nasal, siendo así una de las principales causas de morbilidad aguda en los países industrializados y, a menudo, justifica las ausencias a la escuela y al trabajo [15].

Covid-19

Es una enfermedad respiratoria altamente contagiosa causada por el virus Sars-CoV-2 (Síndrome Respiratorio Agudo Severo) que surgió en Wuhan el 31 de diciembre de 2019. El virus se transmite de persona a persona por vía oral, esto se debe a que cuando tose las gotas que contienen el virus se propagan a su alrededor, de la misma forma que cuando estornuda o habla. Otra forma de propagación de la enfermedad es que una persona sana entre en contacto con la superficie infectada y luego con la boca, la nariz o los ojos. Causa signos y síntomas respiratorios agudos o graves, que generalmente se manifiestan entre 2 a 14 días después de la exposición al coronavirus como: fiebre, tos y dificultad al respirar, en algunos casos se presenta fatiga, dolor muscular, escalofríos, dolor de cabeza, dolor de garganta, náuseas, vómito, diarrea, pérdida del sentido del gusto o el olfato [16].

Rinitis Alérgica

Es un diagnóstico asociado con un conjunto de síntomas que afectan la nariz. Estos síntomas se presentan cuando usted inhala algo a lo que es alérgico, como polvo, caspa o polen. Los síntomas también pueden ocurrir cuando usted consume alimentos a los que es alérgico. Es la inflamación de la mucosa nasal y los síntomas suelen ser los típicos de un resfriado, con picor nasal, estornudos, mucosidad y congestión nasal. Estos síntomas, que en principio se puede pensar que carecen de gravedad, si se mantienen en el tiempo, producen un considerable malestar.

Además, casi siempre se acompañan de conjuntivitis y con mucha frecuencia influyen en el desarrollo de asma bronquial. Muchos estudios muestran que la rinitis afecta más a la vida diaria que el asma bronquial. [17].

1.3.3. Enfermedades de vías respiratorias superior

La vía aérea superior es la región anatómica que se extiende desde la fosa nasal anterior hasta la laringe e incluye las áreas principales, nariz, senos paranasales, oído medio, faringe, laringe y estructuras amigdalinas. En muchos casos, las infecciones de las vías respiratorias superiores son causadas por agentes virales y suelen ser leves y autolimitadas. Sin embargo, en algunas circunstancias, la causa puede ser bacteriana, en cuyo caso el tratamiento antibiótico precoz y correcto puede dar lugar a una adecuada y rápida respuesta clínica [18].

Las infecciones de las vías respiratorias superiores conducen a un mayor consumo de drogas. También identificaron, con cierta frecuencia, el uso inadecuado de antibióticos, lo que contribuye a la rápida aparición de resistencias frente a microorganismos previamente susceptibles, como ocurrió con los neumococos. De acuerdo con los estándares de tratamiento acordados en las guías de práctica clínica, la disponibilidad de pruebas de diagnóstico más rápidas e información de salud de la población más precisa sobre el uso de antibióticos puede contribuir a actividades médicas más eficientes. Las infecciones del tracto respiratorio superior son una de las razones más comunes de visitas en atención medicina general, tanto en niños como en adultos [18].

1.3.4. Desarrollo Web

El desarrollo web se refiere a la construcción, creación y mantenimiento de sitios web. Incluye aspectos como diseño web, publicación web, programación web y base de datos gestión. Además, consiste en utilizar lenguajes de programación como: HTML, CSS, JavaScript, entre otros, para crear programas que sean intuitivos, dinámicos y ágiles.

Un desarrollador web puede participar en el diseño de un sitio web, pero también puede construirlo, este trabajo se tiene en segundo plano pero es el principal para que la web funcione y se mantenga íntegra e impecable, además de que tenga un funcionamiento rápido y un buen desempeño, puede ayudar a mantener y actualizar una base de datos utilizada por un sitio web dinámico [19].

El desarrollo web incluye muchos tipos de creación de contenido web. Algunos ejemplos incluyen páginas web de codificación manual en un editor de texto, crear un sitio web en un programa como Visual Studio Code y actualizar un Blog a través de un sitio web de blogs. En los últimos años, los sistemas de gestión de contenido como WordPress, Wix y Squarespace también se han convertido en medios populares de desarrollo web. Estas herramientas facilitan que cualquiera pueda crear y editar su propio sitio web utilizando un sitio web interfaz [19].

1.3.5. Aplicación móvil

Las aplicaciones móviles se utilizan en teléfonos inteligentes, tabletas, además los usuarios pueden interactuar con la aplicación y realizar todo tipo de actividades para las cuales fue creada la app. Posee una gran adaptabilidad a diferentes dispositivos, permitiendo incrementar los servicios y contenidos para sus usuarios. Estas aplicaciones se instalan en su dispositivo de forma predeterminada y otras se descargan por medio de su conexión a Internet, se dividen en dos categorías: gratuitas y de pago. Hoy en día, las aplicaciones móviles ya forman parte de la vida cotidiana de todas las personas ya que facilitan las actividades y permiten mejorar la comunicación y difusión de la información [20].

1.3.6. Tipos de aplicaciones móviles

Existen tres tipos diferentes de aplicaciones móviles en función de cómo se desarrollan, el uso previsto del código fuente y el framework (herramienta utilizada para desarrollar la aplicación).

Aplicaciones nativas

Tienen características típicas de un dispositivo, están desarrollados específicamente para un sistema operativo y carecen de compatibilidad multiplataforma, están desarrolladas en

base a un lenguaje de programación específico, son capaces de un mejor soporte y fluidez en función del uso de recursos de hardware [20].

Aplicaciones Web

Están desarrolladas en un lenguaje compatible con todos los dispositivos con acceso y conexión a Internet, la programación es más sencilla porque independientemente del sistema operativo del dispositivo objetivo, la aplicación debe ser responsive, es decir, puede que debe adaptarse a cualquier pantalla proyectada [20].

Aplicación Híbridas

Es una combinación de aplicaciones web y nativas que optimizan los mejores recursos de cada una de las mencionadas, permitiéndoles ser utilizadas en más dispositivos ocupando su hardware, por cierto se tiene mejor control y compatibilidad dependiendo del lenguaje de programación, no es necesario estar en una tienda de aplicaciones para utilizar el lenguaje web, muy al contrario de lo que sucede con las aplicaciones nativas [20].

1.3.7. Inteligencia Artificial

La Inteligencia Artificial (IA) es un campo de la informática y la ingeniería que se centra en el desarrollo de sistemas y programas capaces de realizar tareas que, hasta ahora, solo podían ser realizadas por seres humanos. La IA se basa en la creación de algoritmos y modelos matemáticos que permiten a los sistemas computacionales aprender de la experiencia, adaptarse a situaciones cambiantes y tomar decisiones inteligentes en base a los datos [21]. Para su conceptualización se le ha dividido en cuatro partes, divididos en tres argumentos como lo es los procesos mentales, el razonamiento y la conducta.

- **Sistemas que piensan como humanos**

Es la automatización de las actividades que están ligadas al proceso del pensamiento humano, como la toma de decisiones, la resolución de problemas y clasificación [21].

- **Sistemas que actúan como humanos**

Es la investigación de cómo tratar de que un computador pueda hacer una tarea que los humanos la hacen a la perfección hoy en día [21].

- **Sistema que piensan racionalmente**

Es el estudio de los cálculos para realizar el percibir, razonar y actuar [21].

- **Sistema que actúan racionalmente**

La IA está vinculada con el comportamiento de manera inteligente de los aparatos electrónicos [21].

1.3.8. Tipos de Inteligencia Artificial

Aprendizaje automático o Machine Learning

Es un subcampo de la IA que evolucionó a partir del estudio del reconocimiento de patrones y la teoría del aprendizaje computacional en Inteligencia Artificial, tiene como fin resolver problemas muy complejos con la menos inferencia humana, en base del tipo de aprendizaje automático que se ocupe puede brindar mejoras automáticas como tener pequeños ajustes o verificar errores y éxitos del sistema, el aprendizaje automático se divide en: aprendizaje supervisado, aprendizaje sin supervisión y aprendizaje reforzado [22].

1.3.9. Flutter

Es un kit de desarrollo de software con interfaz gráfica open source creado por Google. Es utilizado actualmente en el desarrollo de aplicaciones para Android, iOS, Windows,

Mac, Linux y web a partir de un único código base. El proceso utilizado es la compilación del código en lenguaje Dart a código nativo de cada plataforma [23].

Este framework de estilo reactivo se complementa con un motor de renderizado 2D, widgets listos para utilizar y herramientas de desarrollo. La arquitectura de Flutter se divide en los siguientes componentes:

- Plataforma Dart: para dispositivos móviles se realiza compilación AOT que permite que tenga un rendimiento elevado. Además, permite hot reload durante el desarrollo, es decir, permite realizar cambios y ver su comportamiento sin necesidad de volver a compilar toda la aplicación [20].
- Motor de Flutter: escrito en C++ y con soporte de renderizado de bajo nivel gracias a las librerías de gráficos de Skia de Google. Interactúa con el SDK específico de cada plataforma para implementar gráficos, entrada y salida de ficheros y de red, así como el tiempo de ejecución de Dart [20].
- Widgets específicos para cada plataforma, concretamente widgets de Material Design que implementan el lenguaje de diseño de Google y widgets Cupertino que implementan las guías de interfaz humana de iOS de Apple [20].

1.3.10. Dart

Es un lenguaje orientado a objetos, basado en clases, reflexivo, con recolección de basura autogestionada e imperativo. Tiene una sintaxis muy parecida a C y está influenciado por lenguajes como Java, JavaScript, Erlang, Kotlin, Ruby o Strongtalk [20].

También puede ser compilado a código nativo o a JavaScript, el lenguaje está optimizado para el diseño de interfaces gráficas y cuenta con mecanismos para escribir código basado en eventos y un sistema de tipado flexible. Además, permite la iteración mediante cambios en caliente que se ven reflejados inmediatamente en las aplicaciones. Su rapidez en todas las plataformas y su flexibilidad para escribir tanto el frontend como el backend hacen de este lenguaje una auténtica apuesta ganadora en estos momentos [20].

1.3.11. Firebase

Firestore se trata de una plataforma móvil creada por Google, cuya principal función es desarrollar y facilitar la creación de apps de una forma rápida, con el fin de que se pueda aumentar la base de usuarios y ganar más dinero. La plataforma está subida en la nube y está disponible para diferentes plataformas como iOS, Android y web. Contiene diversas funciones para que cualquier desarrollador pueda combinar y adaptar la plataforma a medida de sus necesidades [24] .

Firestore: Es una base de datos NoSQL, aunque presenta diversas diferencias. Se organiza en forma de documentos agrupados en colecciones, y en ellos se pueden incluir tanto campos de diversos tipos (cadenas de texto, números, puntos geográficos, referencias a la propia base de datos, arrays, booleanos, marcas de tiempo, e incluso objetos propios) como otras subcolecciones [25].

1.3.12. Railway

Railway es una plataforma de implementación donde puede proporcionar infraestructura, desarrollar con esa infraestructura localmente y luego desplegar en la nube. Cuenta con mas de 75 plantillas para poder compilar los diferentes tipos de programas en casi todos los lenguajes de programación [26].

Además, se caracteriza por ser amigable con GitHub ya que con esto brinda mas facilidades a los desarrolladores al momento de querer hacer un deploy en un proyecto. Cuenta con documentación para cualquier tipo de proyecto, ya sea usando una plantilla o subir el código de manera manual.

1.3.13. Metodologías Agiles

Es un tipo de metodología para optimizar los procesos que se venían realizando mediante metodologías tradicionales, su finalidad es demostrar al usuario avances evidentes del proyecto de software, además se genera un mejor vinculo del usuario con el equipo de

desarrollo, de esta manera se puede elaborar el software planificado acorde a lo solicitado por el usuario, este tipo de metodología se basa en:

- Demorar la toma de decisiones
- Planificación adaptativa

1.3.14. Metodologías Ágiles para el desarrollo de aplicaciones móviles

- **Programación Extrema (XP):** Es un tipo de metodología ágil la cual tiene como fin poner más interés en la adaptabilidad que en la previsibilidad, esta pensada para que se desarrolle en un grupo de trabajo de dos a diez desarrolladores, mismos que tienen distintas responsabilidades y su principal objetivo es cubrir todos los requerimientos en el transcurso del desarrollo del proyecto.
- **SCRUM:** La finalidad de esta metodología es que los programadores empiezan de los requerimientos mas generales y debe finalizar el proyecto en menos tiempo que la de un proyecto similar desarrollado antes. En esta metodología se define de manera muy clara procesos y roles y esta diseñada para ir entregando de manera constante avances del software que se esta elaborando, se tiene tres roles:
 - ✓ Scrum Master: Líder del proyecto
 - ✓ Product Owner: Clientes que tienen el rol de dueños del producto
 - ✓ Equipo de programadores: Usualmente está compuesto de cinco a nueve personas.
- **Mobile-D:** Esta metodología se basa en la planificación y la creación de fases, seccionando de esta manera el proyecto en la fase de inicialización se crea un cronograma para planificar las actividades y fases necesarias para el desarrollo del proyecto las cuales tienen un día establecido de planificación y un día de entrega, esta compuesta por las siguientes fases:
 - ✓ Exploración
 - ✓ Inicialización
 - ✓ Fase de Producto

- ✓ Fase de Estabilización
- ✓ Fase de Pruebas

1.4. Objetivos

1.4.1. Objetivo General

Implantar un asistente virtual para la detección de Enfermedades Respiratorias mediante el uso de Inteligencia Artificial.

1.4.2. Objetivos Específicos

- Determinar la sintomatología de las enfermedades respiratorias asociados a: gripe común (rinofaringitis), Covid19 y Rinitis alérgica.
- Seleccionar un algoritmo de aprendizaje de maquina supervisado que se adapte al asistente virtual para detección de afecciones respiratorias.
- Implementar el asistente virtual ocupando un algoritmo que prediga la sintomatología de las enfermedades respiratorias propuestas.
- Integrar el asistente virtual al proyecto de investigación “Sistema electrónico de monitoreo de bioseñales y alertas de distanciamiento social para la prevención de SARS-COV2 aplicando Inteligencia Artificial”.

CAPITULO II.- METODOLOGÍA

2.1. Materiales

Para la recopilación de datos cuya base consiste en buscar información sobre la sintomatología de las enfermedades respiratorias: gripe común, rinitis alérgica y covid-19, se ocupó el método de observación como instrumento de análisis, y se tomó como fuente de referencia la base de datos obtenida del Hospital San Cayetano en la cual se describen los signos y síntomas de las enfermedades propuestas (ver Anexo B).

Además de esta información, se usaron los datos recopilados de una pulsera electrónica, misma que indica la temperatura del paciente, para vincular la aplicación móvil con la pulsera electrónica se utilizó la base de datos Firebase. Los datos mencionados previamente serán utilizados para el entrenamiento de la IA, adicional a los datos obtenidos por los médicos especialistas. La pulsera electrónica es parte del proyecto de investigación “Sistema electrónico de monitoreo de bioseñales y alertas de distanciamiento social para la prevención de SARS-COV2 aplicando Inteligencia Artificial” (ver Anexo C).

Estructura del Dataset

El dataset usado en este proyecto contiene información acerca de los síntomas y signos que caracterizan a las enfermedades respiratorias propuestas (COVID19, Rinitis Alérgica, Gripe Común o Resfriado).

En la figura 1 se puede observar el número de veces que se presenta en el dataset cada uno de los síntomas y signos que se presentan en cada una de las enfermedades. Como se puede observar las sintomatologías con mayor incidencia son dolor de garganta y tos con más de 3500 casos cada uno; por el contrario, el lagrimeo y los ronquidos son los síntomas con menor incidencia con menos de 150 casos cada uno.

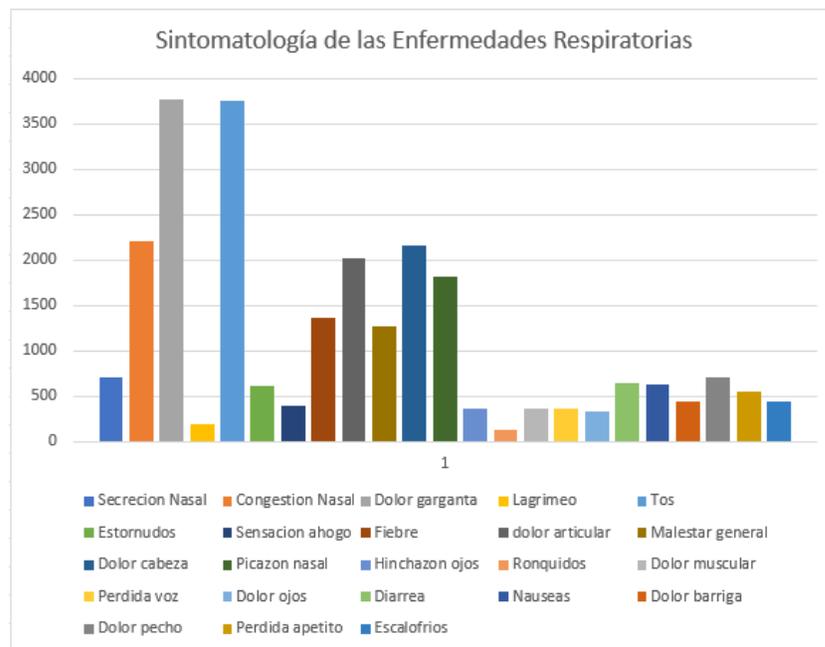


Figura 1. Sintomatología de las enfermedades respiratorias
Elaborado por: El Investigador

En la figura 2 se demuestra que dentro del dataset existen celdas que contienen datos y otras que no; en este caso, se puede concluir que existe una mayor cantidad de celdas vacías que representan al 83% de la información total y el restante corresponde a las celdas llenas con un 17%.



Figura 2. celdas vacías y celdas llenas
Elaborado por: El Investigador

Reemplazo de las celdas vacías con valor 0

En el dataset obtenido del Hospital Básico San Cayetano se han reemplazado las celdas vacías con ceros, ya que no se pueden eliminar columnas debido a que los síntomas del COVID19 utilizan la mayor parte de celdas para que el modelo tenga más certeza puesto que tiene más datos para los predictores.

En la figura 3 se puede visualizar el número de casos en los que se ha presentado cada síntoma dentro de cada una de las enfermedades propuestas. Es decir, en el COVID19 existe una mayor tendencia a presentar dolor de garganta y dolor articular en comparación de síntomas como la secreción nasal y los estornudos que no se presentan frecuentemente en estos casos; por otro lado, en el Resfriado se tiene al dolor de garganta y dolor de cabeza como los síntomas con mayor incidencia frente al lagrimeo y secreción nasal que no se presentan frecuentemente en estos casos; finalmente, en la Rinitis, se presentan con mayor frecuencia síntomas como la picazón nasal y la tos frente a los ronquidos y estornudos que no tienen una incidencia relevante en esta enfermedad.

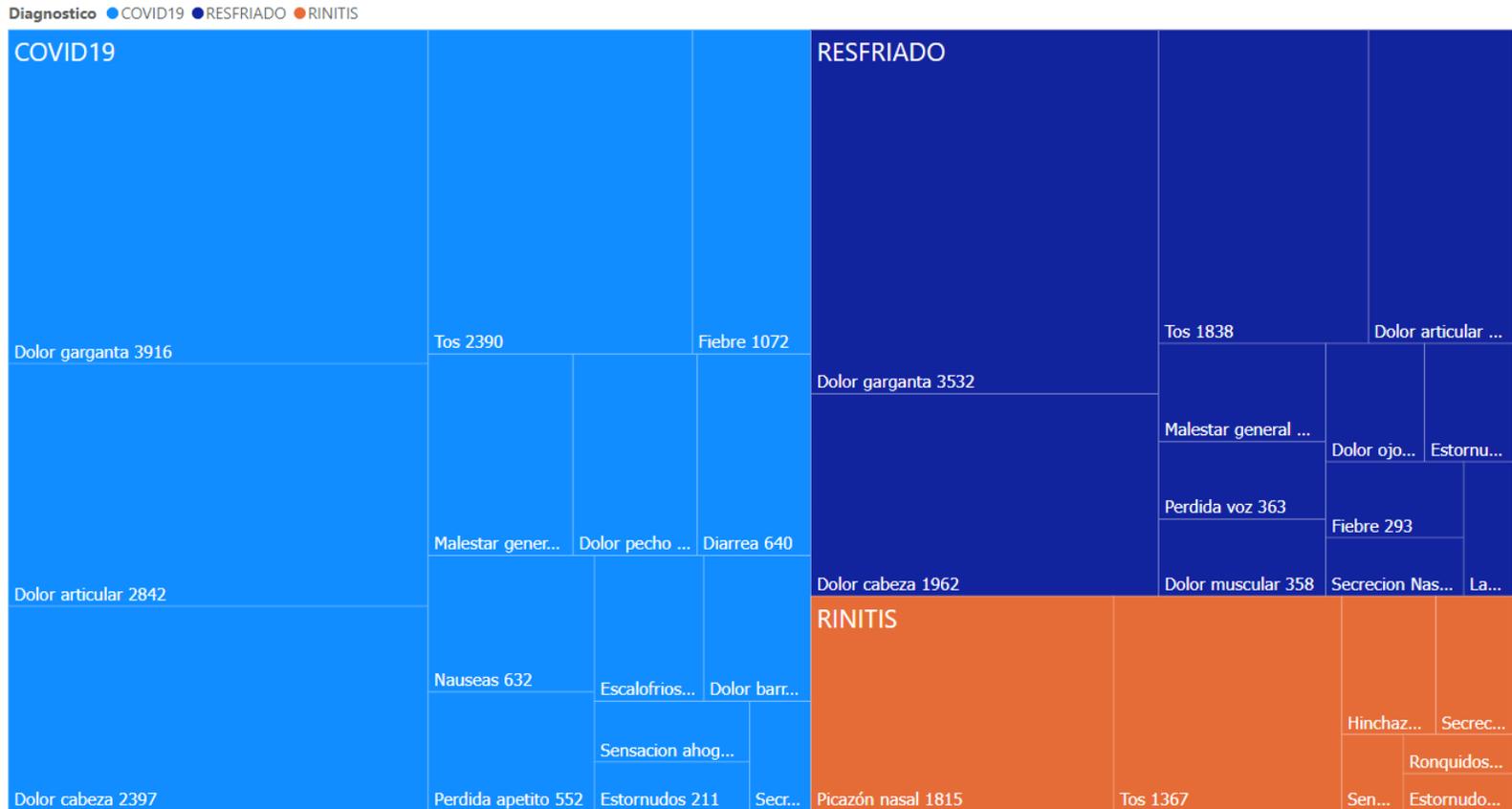


Figura 3. Incidencia de la sintomatología en cada enfermedad respiratoria

Elaborado por: El Investigador

En la figura 4 se puede visualizar el género (masculino, femenino) y el nivel de afectación de cada uno por el síntoma dolor de garganta. En este caso, se puede observar que 2178 hombres no sintieron dolor de garganta, 653 sintieron un dolor leve, 641 un dolor moderado y 561 sintieron un dolor de garganta fuerte; por otro lado, en el caso de las mujeres, 2059 no sintieron dolor, 637 sintieron un dolor leve, 653 tuvieron un dolor moderado y 629 sintieron un dolor de garganta fuerte.

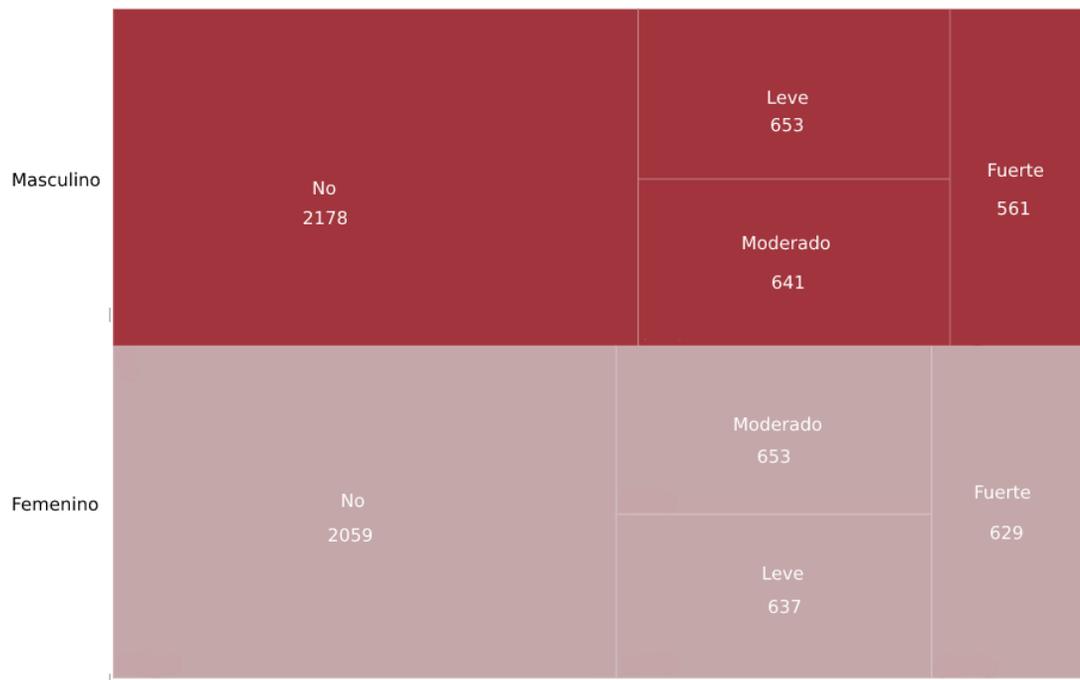


Figura 4. Comparativa entre dolor de garanta y género

Elaborado por: El Investigador

En la figura 5 se puede visualizar el género (masculino, femenino) y el nivel de afectación de cada uno por el síntoma dolor articular. En este caso, se puede observar que 3012 hombres no sintieron dolor en las articulaciones, 385 sintieron un dolor leve, 213 un dolor moderado y 423 sintieron un dolor articular fuerte; por otro lado, en el caso de las mujeres, 2973 no sintieron dolor, 375 sintieron un dolor leve, 234 tuvieron un dolor moderado y 396 sintieron un dolor articular fuerte.

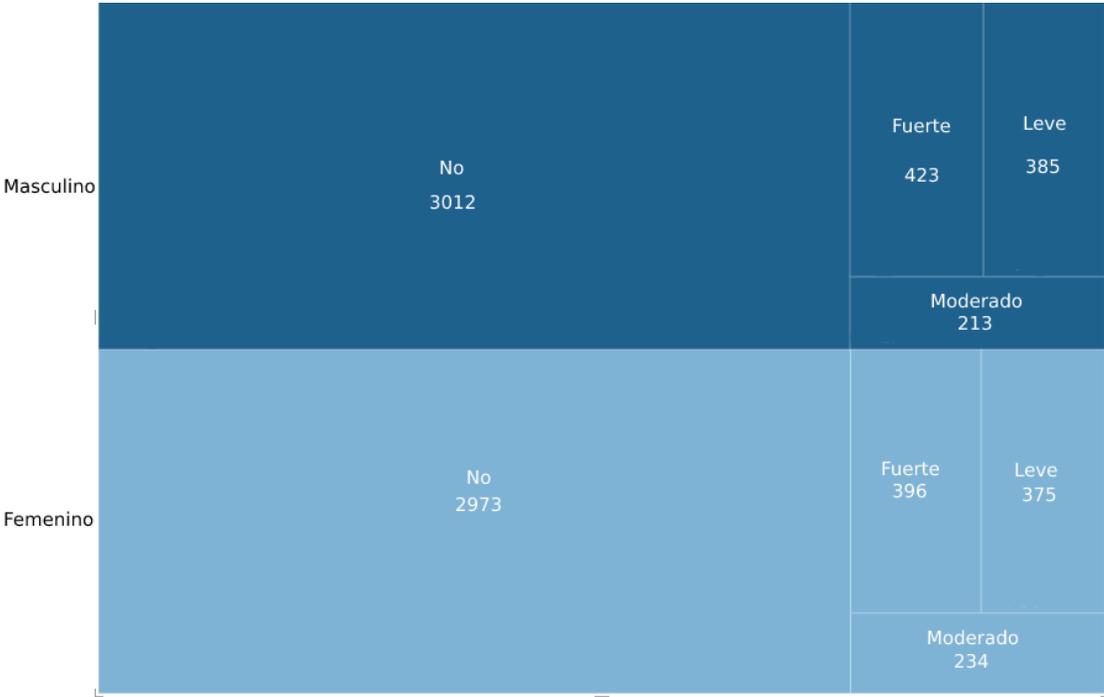


Figura 5. Comparativa Dolor Articular y Género

Elaborado por: El Investigador

En la figura 6 se puede visualizar el género (masculino, femenino) y el nivel de afectación de cada uno por el síntoma dolor de cabeza. En este caso, se puede observar que 2936 hombres no sintieron dolor de cabeza, 348 sintieron un dolor leve, 378 un dolor moderado y 371 sintieron un fuerte dolor de cabeza; por otro lado, en el caso de las mujeres, 2915 no sintieron dolor, 352 sintieron un dolor leve, 343 tuvieron un dolor moderado y 368 sintieron un fuerte dolor de cabeza.



Figura 6. Comparativa entre dolor de cabeza y género

Elaborado por: El Investigador

En la figura se puede visualizar el género (masculino, femenino) y el tipo de afectación de cada uno por el síntoma tos. En este caso, se puede observar que 2103 hombres no tuvieron tos, 967 presentaron tos seca y 963 tuvieron tos con flema; por otro lado, en el caso de las mujeres, 2159 no presentaron tos, 936 tuvieron tos seca y 883 presentaron tos con flema.

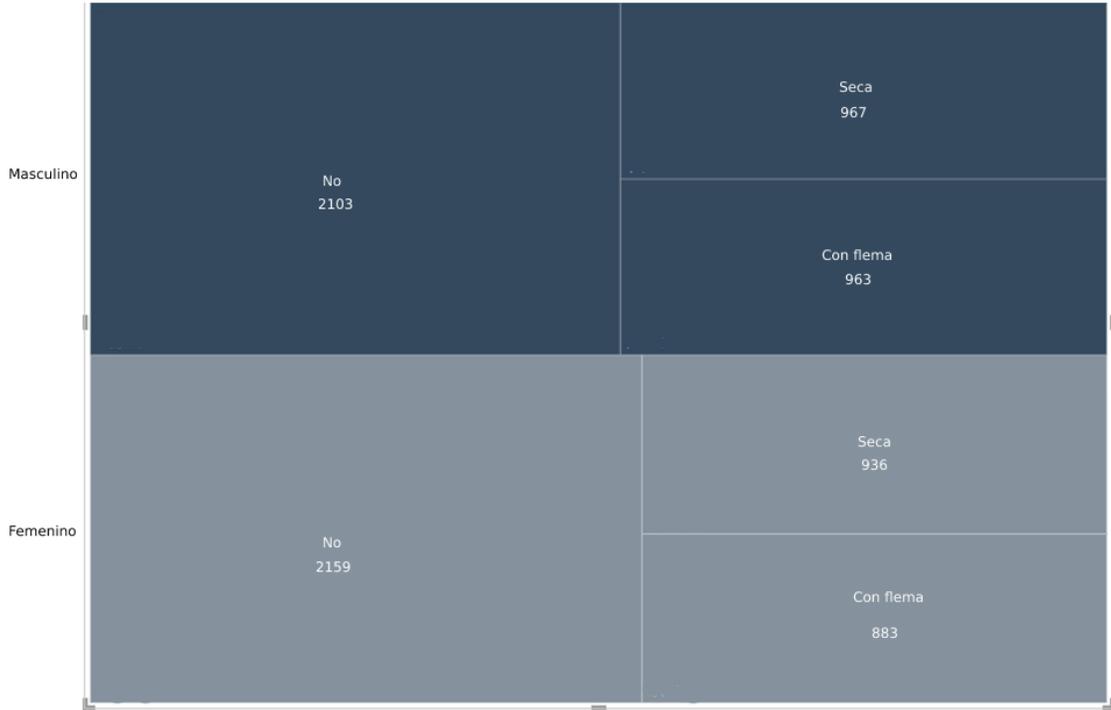


Figura 7. Comparativa entre género y tos.

Elaborado por: El Investigador

2.2. Métodos

2.2.1. Modalidad de Investigación

Modalidad bibliográfica dado que se utilizarán fuentes primarias de información tales como revistas científicas, libros, tesis entre otros, dando las bases teóricas para la construcción del marco teórico y la contextualización del asistente médico virtual.

Modalidad aplicada porque se fundamentará en la resolución del problema por medio del conocimiento adquirido en la carrera para desarrollar un software que brindará ayuda al usuario mediante el uso del asistente médico virtual.

Investigación Documental

Se realizó un estudio documental puesto que es una investigación con médicos especialistas con altos conocimientos de las sintomatologías lo cual ayudó a realizar la recolección de datos de las enfermedades respiratorias: gripe común, rinitis alérgica y covid-19.

Investigación Experimental

Se realizó investigación experimental ya que se tuvo que realizar verificaciones de la validez del algoritmo de la IA en base a la toma de muestras con los estudiantes de la Universidad Técnica de Ambato para realizar una valoración medica de las enfermedades respiratorias: gripe común, rinitis alérgica y covid-19.

Investigación Aplicada

Se realizó una investigación aplicada debido a los conocimientos adquiridos en el transcurso de la formación académica; además, se realizaron investigaciones en las bases de datos indexadas de la Universidad.

Investigación Bibliográfica-documental

Se realizó un estudio bibliográfico documental para realizar una comparativa y escoger el algoritmo que más se ajuste a la creación de un asistente virtual (chatbot), de la misma forma para la selección de un framework de desarrollo que mejor se adapte a la Inteligencia Artificial.

2.2.2. Población y Muestra

En el proyecto se trabajará con una población determinada, que está formada por la comunidad estudiantil de la Universidad Técnica de Ambato. Con la muestra establecida se verificará la validez del asistente médico virtual a partir del uso de la aplicación por parte de los miembros que conforman esta muestra. La comunidad estudiantil supera los 100 elementos, es por lo que se requiere el cálculo de la muestra.

El cálculo del tamaño de la muestra tiene un margen de error del 9%, con un nivel de confianza del 95%.

- Tamaño de la muestra = n
- Población N (población total de la Universidad Técnica de Ambato) = 18596
- Nivel de confianza (Z) = 0.95 equivalente a 1.96
- Error de muestreo (e) = 0.09
- Desviación estándar (s) = 0.5

$$n = \frac{(18596)^2(1.96)^2}{(18596 - 1)^2(0.09)^2 + (0.5)^2(1.96)^2}$$

Tabla 1. Población de estudio
Elaborado por: El Investigador

Población Médicos Especialistas	Número	Muestra (n)
Comunidad Estudiantil	18596	118

2.2.3. Recolección de Información

Para recolectar la información que posteriormente sirvió para entrenar el algoritmo de Inteligencia Artificial, se usó la base de datos brindados del Hospital Básico San Cayetano (ver Anexo B), de esta manera se pudo identificar los síntomas y signos de los casos positivos para las enfermedades respiratorias: gripe común, rinitis alérgica y covid-19. Además de los datos obtenidos de la pulsera electrónica perteneciente al proyecto de investigación “Sistema electrónico de monitoreo de bioseñales y alertas de distanciamiento social para la prevención de SARS-COV2 aplicando Inteligencia Artificial” (ver Anexo C).

2.2.4. Procesamiento y análisis de datos

En base a las modalidades de investigación y los datos obtenidos en la recolección de información se realizó el procesamiento y análisis de datos con base en:

- Recopilación de la información validada de la Base de Datos del Hospital Básico San Cayetano.
- Interpretación de los datos obtenidos para obtener los resultados.

Tabla 2. Segmentación de sintomatología de las enfermedades respiratorias.

Elaborado por: El Investigador

Enfermedad	Síntomas
Gripe común (Resfriado)	Dolor de garganta Dolor de cabeza Tos seca Dolor articular Malestar general Perdida de voz
COVID-19	Dolor de garganta Dolor articular Dolor de cabeza Tos con flema Dolor de pecho Pérdida de apetito escalofríos
Rinitis Alérgica	Picazón nasal Tos seca Estornudos Sensación de ahogo Hinchazón de ojos Ronquidos

- Planteamiento de una solución a la problemática.
- Investigación y comprobación de la validez de los algoritmos para el diagnóstico de las enfermedades respiratorias gripe común, rinitis alérgica y covid-19.
- Análisis y comprobación de la validez del algoritmo mediante la implementación en el proyecto.

CAPITULO III.- RESULTADOS Y DISCUSIÓN

3.1. Análisis y Discusión de los Resultados

La implementación del prototipo de un sistema móvil para el diagnóstico de enfermedades respiratorias: Gripe Común (Rinofaringitis), Covid19 y Rinitis Alérgica, para que realice de manera virtual la toma de signos y síntomas a los pacientes, esto permitirá brindar un diagnóstico en base a los parámetros detectados y con ello se dirijan a un centro médico o descarten el posible contagio.

3.1.1. Algoritmos para el desarrollo de asistentes virtuales

Para el diagnóstico de las enfermedades respiratorias existe varios algoritmos que permiten brindar una ayuda en el proceso de toma de decisiones mediante la clasificación de texto tales como:

Árbol de decisión: Es similar a un diagrama de flujo el cual esta compuesto por nodos y cada nodo tiene una característica o atributo, es un método de tipo aprendizaje supervisado, usado principalmente para los modelos predictivos ya que tiene mucha precisión al momento de seleccionar en base a una o mas variables predictoras continuas o características únicas la decisión.

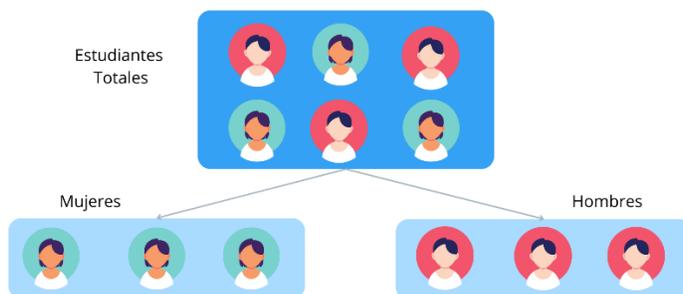


Figura 8. Árbol de Decisiones

Elaborado por: El Investigador

Random Forest: Es un método de aprendizaje por conjuntos, que es capaz de clasificar en base a características o variables predictoras, también es útil en caso de tener valores perdidos, valores atípicos, además fortifica los modelos débiles, en base a la unión para formar un modelo mucho mas robusto y compuesto.

En este método se ejecutan varios arboles de decisión en lugar de uno por uno, y su funcionalidad se toma en base a la mayor cantidad de aciertos a una opción o en otras palabras la opción mas botada es la respuesta.

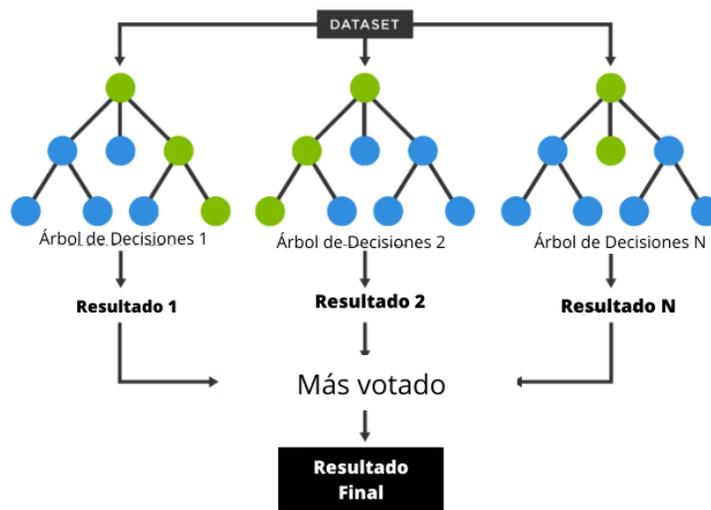


Figura 9. Random Forest

Elaborado por: El Investigador

Support Vector Machine (SVM): Es un método de clasificación, el cual tiene una evolución considerable ya que empezó siendo un problema de clasificación binaria y actualmente puede resolver múltiples clasificaciones, destaca ya que tiene muchas situaciones probables, tiene su fundamentación en Maximal Margin Classifier el cual define al hiperplano como un subespacio plano con dos dimensiones en las cuales abarca una dimensión ya que es una recta que separa los datos.

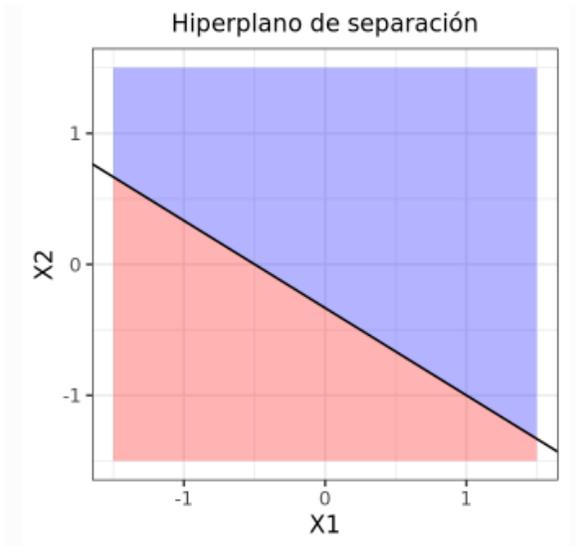


Figura 10. Support Vector Machine
Elaborado por: El Investigador

3.1.2. Comparación entre algoritmos de asistentes virtuales

Tabla 3. Tabla comparativa de algoritmos para asistentes virtuales

Elaborado por: El Investigador

Algoritmos	Características			
	Precisión	Exactitud	Tiempo de computación	Clasificación de datos mezclados
Random Forest	$\frac{VP}{(VP + FP)}$ <p>0.91</p>	$\frac{(VP + VN)}{(VP + VN + FP + FN)}$ <p>0.89</p>	No se genera un tiempo de computación excesivo, ya que el modelo tolera hasta 100 árboles de manera estable n_estimators	Se crea un dataset nuevo de manera aleatoria (100) para crear los árboles con menos características
Árbol de Decisiones	$\frac{(VP + VN)}{(VP + FP + VN + FN)}$ <p>0.87</p>	$\frac{VP}{(VP + FP)}$ <p>0.85</p>	El modelo es muy simple y no representa un consumo de recursos del computador	Se toma como raíz el nodo que tenga menos impureza (Gini Impurity)
Support Vector Machine	$\frac{TP}{(TP + FP)}$ <p>0.90</p>	$\frac{(TP + TN)}{(TP + TN + FP + FN)}$ <p>0.88</p>	Si, requiere mucho tiempo ya que se analiza el número de parámetros y el número de datos de entrenamiento.	Lo hace en base a como se ajuste el parámetro C de manera manual

3.1.3. Selección del Algoritmo para el desarrollo de asistentes virtuales

A partir de la comparación realizada entre los algoritmos y las características para generar diferencias entre cada uno, se ha decidido que lo óptimo es utilizar el algoritmo Random Forest debido que tiene valores de 0.91 y 0.89 en precisión y exactitud respectivamente y se puede evaluar de mejor manera el modelo, además tiene un procesamiento de datos similar a un árbol de decisiones, pero con una estructura más compleja ya que crea de manera aleatoria árboles de decisiones y brinda la opción adecuada a seguir con base en los resultados de dichos árboles. Además, este modelo tiene la posibilidad de manejar predictores numéricos y categóricos y no necesita tener un dataset estructurado pues toma los datos de manera aleatoria para ajustarlos al modelo, tiene también una buena escalabilidad y puede aplicarse a datasets que contengan un gran número de datos.

Para adaptar a la problemática los algoritmos seleccionados, se llevó a cabo un proceso de codificación en el cual se realiza una estructura formada por el entrenamiento que representan al 70% de la totalidad y otros archivos destinados al testeo que representan al 30% de la totalidad del dataset proporcionado por el Hospital Básico San Cayetano. Para realizar este entrenamiento se genera una validación cruzada, una matriz de confusión y se crean los archivos físicos del modelo para posteriormente realizar el testeo.

3.1.4. Frameworks para el desarrollo de aplicaciones móviles

Para el desarrollo de aplicaciones móviles existen varios frameworks que permiten hacerlas de manera nativa ya sea en Android o iOS que son los sistemas operativos más populares y usados actualmente, en base a su popularidad, ventajas de uso se ha seleccionado los siguientes:

- **Flutter:** Es un framework elaborado por Google para generar aplicaciones nativas multiplataforma (Ios y Android), en base a un código único de programación permite crear y diseñar interfaces móviles.

El lenguaje que ocupa Flutter para su codificación es Dart, el cual también fue creado por Google, es un lenguaje orientado a objetos y fuertemente tipado, además que tiene herramientas integradas para una mejor experiencia y comodidad al momento de programar.

- **React Native:** Es un framework que permite realizar aplicaciones nativas basadas en JavaScript y ReactJS. Permite convertir los elementos ya realizados de manera nativa para la interfaz de Android y Ios. Una de sus ventajas es que permite reducir la espera al realizar un cambio en tiempo real y ver reflejado en el aplicativo.
- **Ionic:** Es un SDK de código abierto para elaborar aplicaciones híbridas basadas en CSS, HTML Y JS, permite un diseño limpio y funcional de sus interfaces, además es compatible con Angular, React y Vue. Tiene como características principales su versatilidad debido a que tiene elementos para facilitar la personalización.

3.1.5. Tabla Comparativa de Frameworks para aplicaciones móviles

Tabla 4. Tabla comparativa de frameworks para aplicaciones móviles

Elaborado por: El Investigador

	Flutter	Ionic	React Native
Lenguaje de Programación	Dart	TypeScript/JavaScript	TypeScript/JavaScript
Rendimiento	No requiere transiciones o conexiones nativas para desarrollar en entorno nativo, brinda facilidades para generar una experiencia nativa de alta calidad para el diseño de la interfaz.	Crea aplicaciones híbridas multiplataforma, para compilar de manera nativa necesita servicios de terceros.	Permite crear aplicaciones con componentes nativos lo cual permite dar sensaciones que es una aplicación nativa.
Popularidad	Es un nuevo framework que en poco tiempo ha	Es muy popular entre los desarrolladores de aplicaciones móviles	Tiene mucha popularidad debida que grandes empresas

	obtenido mucha acogida y grandes empresas tales como: Alibaba, Google, Hamilton Musical, entre otros.	y lo ocupan empresas como: MarketWatch, Pacifica, Sworkit, Nationwide.	ocupan dicho framework tales como: Facebook, Instagram, Pinterest, Skype, Tesla, Uber entre otros.
Comunidad	En poco tiempo ha generado un revuelo en redes sociales lo cual ha hecho que tenga una comunidad cada vez mas grande con la ayuda de Google, además que generó un incentivo a los desarrolladores a aprender Dart y a conocer mas a profundidad su framework para que puedan crear aplicaciones nativas.	Se desarrollo hace mas de 6 años lo cual hace que cuente con un fuerte apoyo de la comunidad en base a resolución de problemas o errores que se presente.	Cuenta con una comunidad importante enfocada en el soporte para el desarrollo de aplicaciones móviles

3.1.6. Selección del framework para el desarrollo de la aplicación móvil

En base a la comparación realizada de los frameworks de aplicaciones móviles se ha optado por el uso de Flutter debido que brinda mas beneficios tales como un menor coste de procesamiento del computador a la hora de compilar e instalar el aplicativo en el celular, además tiene menos tiempo de demora al momento de generar un cambio en tiempo real sin perder el estado actual (hot reload), además que su lenguaje de programación Dart es orientado a objetos, lo que quiere decir que se puede reutilizar el código, cabe destacar que las aplicaciones desarrolladas con Flutter tienen un diseño limpio de la interfaz de usuario y brinda sensaciones de ser una aplicación nativa. Brinda

facilidades al momento de compilar el proyecto debido que maneja un código único de programación y ocupa los diferentes compiladores para ejecutar el proyecto en iOS y en Android.

3.1.7. Metodología

Para el desarrollo de un proyecto es necesario tener una metodología la cual se va a usar en base a las necesidades y objetivos que se plantee. Es un grupo de técnicas y pasos las cuales permiten desarrollar un proyecto, la finalidad de aplicar una metodología a un proyecto de creación o desarrollo de software es mejorar la probabilidad de éxito del proyecto, existen varias metodologías que se subdividen en base a las necesidades y requerimientos.

Para el presente proyecto se ha determinado que se necesita una metodología para el desarrollo del aplicativo móvil por lo que hay que realizar un análisis de una comparativa entre que metodología ágil es la mas apropiada en base a los requerimientos y objetivos planteados.

Tabla comparativa de las Metodologías Ágiles para el desarrollo de aplicaciones móviles

Tabla 5.Tabla comparativa de metodologías ágiles

Elaborado por: El Investigador

	XP	SCRUM	MOBILE-D
Tipo de Proyecto	Proyectos medianos y pequeños	Es aplicable a todo tipo de proyecto	Proyectos pequeños
Tipo de trabajo en equipo	Se puede asignar de 2 a 10 desarrolladores.	Se basa en la asignación de roles para distribuir actividades.	Se crean pequeños grupos de trabajo para simplificar las actividades.
Documentación	Tarjetas CRC	Product backlog	Storycards

		Sprint backlog Burndown chart Definition of done Definition of Ready	
Tiempo de producción	Proyectos de larga duración.	Proyectos de poca duración.	Proyectos de poca duración.

3.2. Metodología Mobile -D

3.2.1. Fase de Exploración

En esta primera fase se define la planificación, el alcance que tendrá el desarrollo del software, definir el grupo de interés por parte del cliente y del grupo de desarrollo, además de estimar detalles y la funcionalidad hacia donde se va a llegar, las actividades a desarrollarse en esta fase son las siguientes:

- Definir objetivos del software a desarrollarse
- Definir el alcance del aplicativo móvil para el desarrollo del Asistente Medico Virtual
- Planificación del proyecto

Definición de las partes interesadas

- **Empresa:** Universidad Técnica de Ambato.
- **Usuarios de la aplicación:** Estudiantes de la Universidad Técnica de Ambato.
- **Desarrollador:** Autor del presente proyecto.

Definición de alcance

- Se establece los requisitos del proyecto y en base a ello se establece una planificación para el cumplimiento.
- Los datos obtenidos han sido otorgados del Hospital Básico “San Cayetano” los cuales se procesarán por medio de una Api.

Limitaciones

- El usuario debe tener un dispositivo móvil con S.O Android 9.0 o superior.
- El usuario debe tener acceso a internet.

Definición del Proyecto

Para el desarrollo del proyecto se han ocupado las siguientes herramientas tecnológicas:

- **Flutter:** Framework para el desarrollo del aplicativo móvil.
- **Firebase:** Base de datos no relacional la cual guarda los datos registrados a los pacientes por medio de la aplicación móvil y pulsera electrónica.
- **Api Node JS:** Es el método en el cual se genera la interacción entre la app con el dispositivo generando una separación entre el cliente y el servidor además que se ha alojado el algoritmo establecido para la comparativa y se genere la predicción esperada.

3.2.2. Fase de Inicialización

En esta fase se genera la configuración del proyecto y se prepara los recursos listados en la fase anterior, se determina complejidad en ciertas fases del proyecto que pueden tener una complejidad superior al resto, al igual que se genera el desarrollo del proyecto y su posterior publicación.

Configuración

La configuración física se dio por medio de la instalación de los elementos para el desarrollo del aplicativo móvil, en este caso se ocupa Flutter como framework de desarrollo del Front End usando el editor de código Visual Studio Code debido a que brinda una mayor facilidad al momento de escribir el código que abrir otro entorno de programación, y para el desarrollo y configuración del Back End se ocupó la API, además

que los datos otorgados por la Clínica San Cayetano junto a los tomados a nuevos pacientes están alojados en Firebase.

Diseño de la arquitectura del proyecto

La arquitectura planteada, los estudiantes de la Universidad Técnica de Ambato pueden acceder al aplicativo y poder realizarse un diagnostico médico. La petición pasa del usuario al aplicativo móvil luego por internet por medio de una petición HTTP a la API donde se aloja el algoritmo entrenado y se brinda un resultado en base a los datos ingresados.

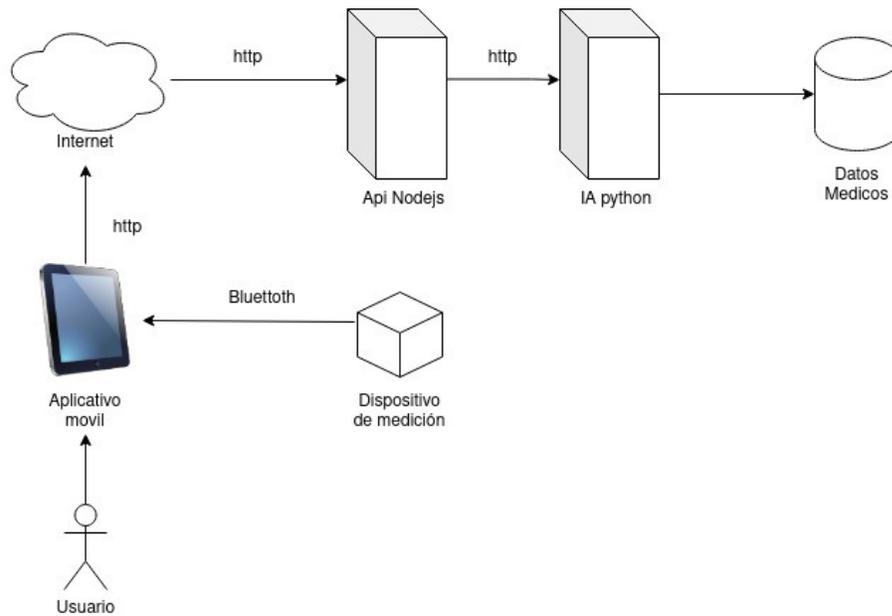


Figura 11. Arquitectura del proyecto

Elaborado por: El Investigador

Estructura de la Base de Datos

Para el desarrollo del proyecto se implementará la Base de Datos de tipo NO SQL en la plataforma Firebase que es de propiedad de Google, debido a la facilidad que existe en

base a documentación, tutoriales y la fácil implementación que representa al momento de vincular un proyecto móvil a la plataforma, Firebase mantiene una estructura en base a los módulos a implementar en el proyecto, en este caso se va a ocupar Firestore Database para almacenar los datos que van a ser obtenidos al momento de tomar un diagnóstico a un paciente.

Colección

Una colección es la estructura de la tabla la cual consta con campos y tipo de datos en los cuales se van a crear documentos para almacenar las tuplas que se obtenga mediante el uso del aplicativo

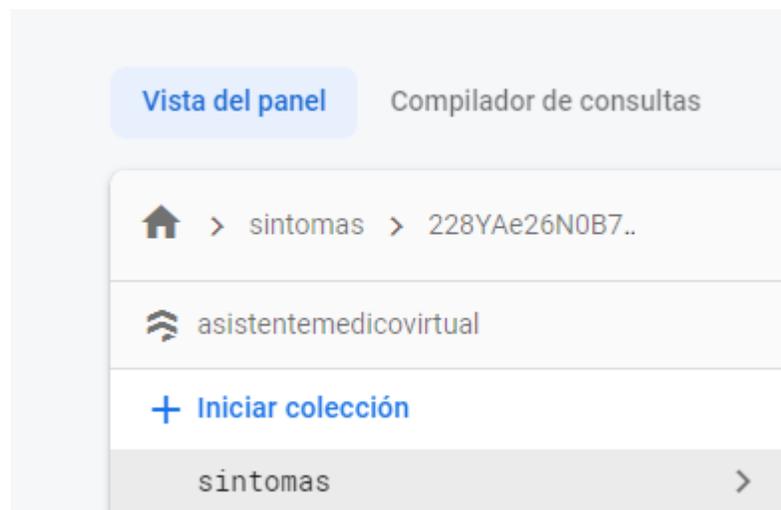


Figura 12. Colección Base de Datos

Elaborado por: El Investigador

Documentos

Los documentos son la tupla de datos que se han obtenido del aplicativo

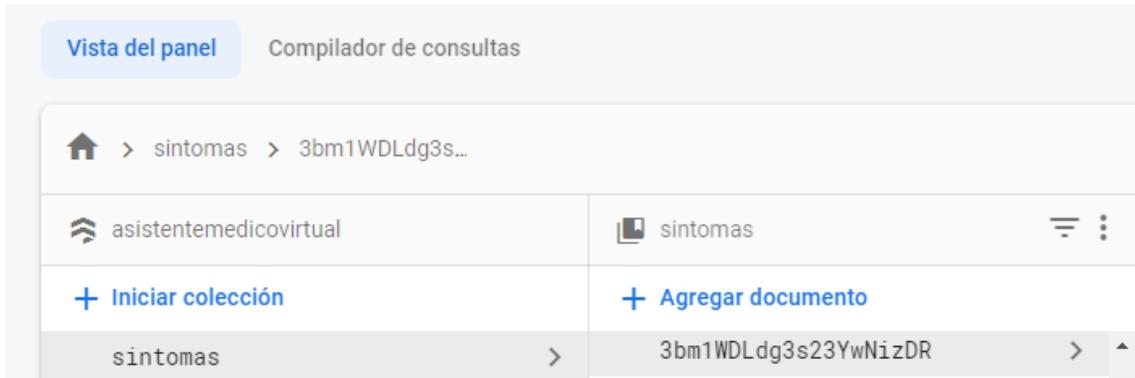


Figura 13. Documentos Base de Datos

Elaborado por: El Investigador

Datos obtenidos en la Base de Datos

Un ejemplo de la estructura que maneja Firebase se muestra en la figura 7, se muestra la estructura de datos como se obtiene del aplicativo y que tipo de dato tiene cada valor en base a la tupla obtenida por medio del diagnóstico realizado con el aplicativo, Firebase maneja cada dato como objeto independiente

3bm1WDLdg3s23YwNizDR

+ Iniciar colección

+ Agregar campo

Congestion Nasal: 1
Diarrea: 0
Dolor barriga: 0
Dolor cabeza: 2
Dolor garganta: 2
Dolor muscular: 0
Dolor ojos: 0
Dolor pecho: 0
Escalofrios: 0
Estornudos: 1
Fiebre: 0
Genero: 1
Hinchazon ojos: 1
Lagrimeo: 0
Malestar general: 1
Nauseas: 0
Perdida apetito: 0
Perdida voz: 0
Picazon nasal: 1
Ronquidos: 0
Secrecion Nasal: 1
Sensacion ahogo: 0
Temperatura: "37.6"
Tos: 1
diagnostico: "RINITIS"
dolor articular: 1

Figura 14. Ejemplo Base de Datos

Elaborado por: El Investigador

Requerimientos

Los estudiantes tendrán acceso al aplicativo del asistente médico virtual, por lo cual podrán realizarse un análisis médico para saber si son o no portadores de las enfermedades respiratorias: gripe común, rinitis alérgica, COVID -19.

Funcionalidades establecidas

En base a los requerimientos y a la base de datos otorgada por el Hospital Básico San Cayetano se ha establecido la siguiente funcionalidad, diagnóstico de las enfermedades en base a las respuestas brindada en cada pregunta planteada, una vez finalizado las preguntas le aparecerá el resultado obtenido con un mensaje en pantalla de la enfermedad determinada.

Diseño de las ventanas del aplicativo móvil

Pantalla Principal

La pantalla principal cuenta con un mensaje de bienvenida que explica como es el funcionamiento del asistente medico virtual y cuales son las enfermedades que esta programado para predecir y como es la funcionalidad del mismo, el cual detalla que en base a las opciones que se tome de las preguntas brindará una respuesta acertada y un botón que ingresa a las preguntas establecidas.



Figura 15. Pantalla de Inicio del asistente médico virtual

Elaborado por: El Investigador

Pantalla de Preguntas

La pantalla secundaria consta de las preguntas que se han establecido como prioritarias para la detección de las enfermedades respiratorias propuestas las cuales van apareciendo en orden con las respectivas opciones, entre las cuales son opciones de si y no y en preguntas de intensidad de dolor o de agravio son opciones como: leve, moderado y fuerte.

El asistente médico virtual brinda la opción al usuario de revisar las opciones marcadas y por ello tiene flechas de izquierda y derecha para poder avanzar o retroceder en el

diagnóstico, esto con el fin que si siente que un síntoma lo puso mal de manera involuntaria pueda corregir sin tener que empezar de cero.

Pantalla de Primera Pregunta

La pantalla como muestra la figura 9, el Asistente Médico Virtual presenta la primera pregunta la cual tiene como opciones dos valores del sexo de la persona (Masculino o Femenino).



Figura 16. Pantalla de Primera Pregunta

Elaborado por: El Investigador

Pantalla de quinta pregunta

Como se muestra en la figura 10, se puede ver la quinta pregunta del aplicativo la cual tiene dos opciones (Si o No).

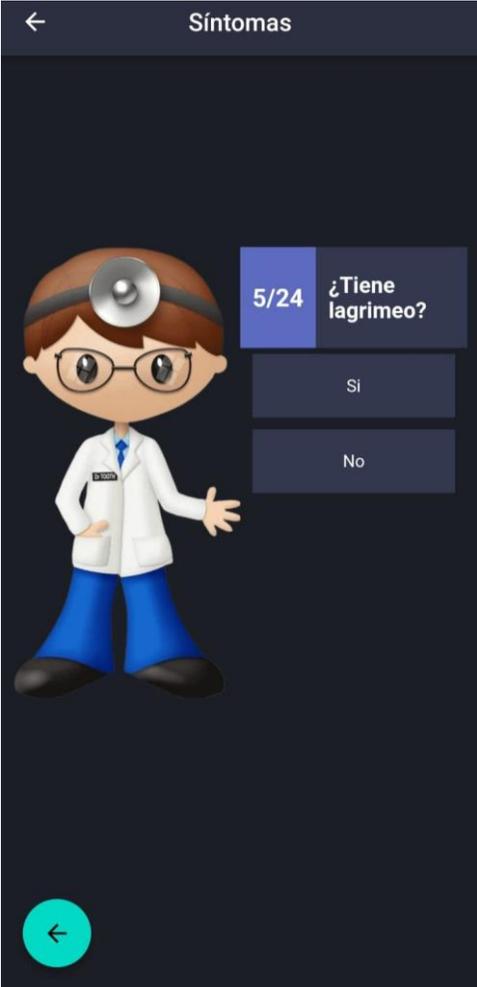


Figura 17. Pantalla de la quinta pregunta
Elaborado por: El Investigador

Pantalla de la decima pregunta

Como se muestra en la figura 11, se puede ver la quinta pregunta del aplicativo la cual tiene tres opciones (Leve, Moderado y Fuerte) categorizando de esta manera la intensidad del dolor.



Figura 18. Pantalla numero 15 del app

Elaborado por: El Investigador

Pantalla de la pregunta veinte

Como se muestra en la figura 12, se puede ver la quinta pregunta del aplicativo la cual tiene dos opciones (Si o No).

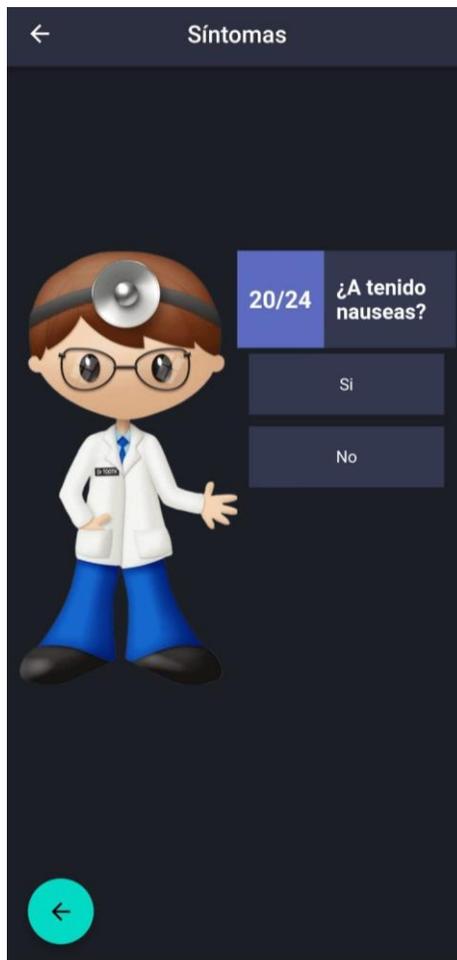


Figura 19. Pregunta numero 20 del app

Elaborado por: El Investigador

Pantalla de ultima pregunta

En la última pantalla de las preguntas consta con un botón de flecha hacia la izquierda para brindar una facilidad en caso de que el usuario quiera comprobar que los síntomas que ingreso son los correctos o si tiene alguna duda sobre alguna pregunta en específico y quiera comprobar la opción elegida, además cuenta con una flecha para enviar los datos ingresados por el usuario y mediante la api se realice el diagnóstico.

Además, cuenta con una flecha en la parte superior que al usuario le permite volver a la pantalla principal, esta función ha sido implementada para brindar al usuario mas comodidades.

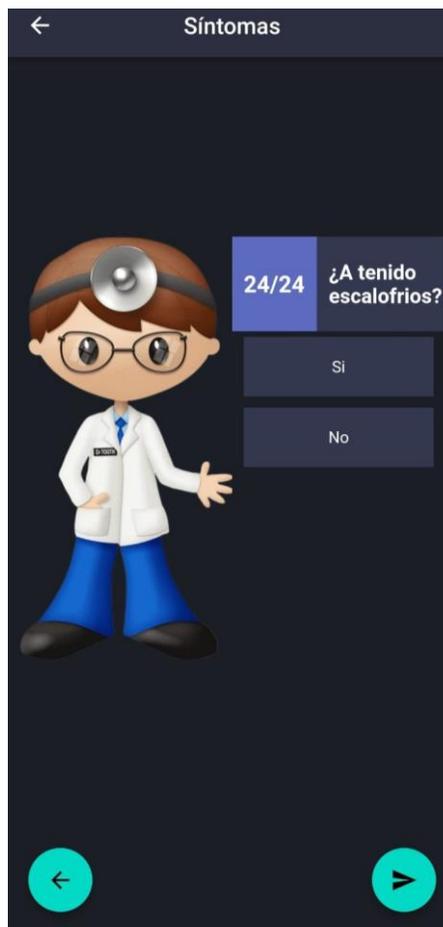


Figura 20. Pantalla de última pregunta
Elaborado por: El Investigador

Pantalla de Diagnóstico

En esta pantalla se muestra el mensaje del diagnóstico el cual ha sido obtenido por medio de las opciones que ha elegido el usuario en las preguntas planteadas, el mensaje varía en base a las opciones elegidas ya que cada enfermedad cuenta con preguntas que son fundamentales para diferenciar una enfermedad de otra, entre las opciones de mensajes que puede salir como mensaje de diagnóstico son: COVID19, GRIPE COMÚN Y RINITIS ALERGICA.



Figura 21. Pantalla de Diagnóstico
Elaborado por: El Investigador

Fase de Producto

En esta fase se comienza con la codificación del proyecto en base a los requerimientos y funcionalidades previamente establecidas.

Aplicativo Móvil

Mediante el uso de Visual Studio Code y el framework de Flutter se ha desarrollado el aplicativo móvil. Partiendo de una estructura de proyecto limpia se direcciona toda la programación hacia el archivo main el cual permite la ejecución, el proyecto consta de tres partes del proyecto como principal las carpetas api y pages y la carpeta widgets que se han creado para el diseño y funcionalidad del aplicativo.

API

La figura 22 se muestra la función `predecirEnfermedad` para obtener una predicción de la aplicación en base a los síntomas ingresados.

```
//se crea una funcion que recibe una lista de sintomas y retorna una prediccion
Future<String> predecirEnfermedad(List<String?> sintomas) {

  Map<String, dynamic> request = {
    'sintomas': sintomas,
  };

  return http.post('predecir', jsonEncode(request))
    .then((resp) {
      var jsonResp = jsonDecode(utf8.decode(resp.bodyBytes));
      if (jsonResp['estado']['codigo'] != 0) {
        throw jsonResp['estado']['mensaje'];
      }
      return jsonResp['payload'].toString();
    }).catchError((ex){
      throw ex;
    });
}
```

Figura 22. Función Predecir Enfermedad

Elaborado por: El Investigador

La figura 23 indica como se obtiene el ultimo valor registrado por medio del sensor (ver anexo C) en la base de Firebase.

```
Future<num?> GetTemperatura() {  
  return http.get('temperatura').then((resp) {  
    var jsonResp = jsonDecode(utf8.decode(resp.bodyBytes));  
    if (jsonResp['estado']['codigo'] != 0) {  
      throw jsonResp['estado']['mensaje'];  
    }  
    var temperatura = num.tryParse(jsonResp['payload'].toString());  
    return temperatura;  
  }).catchError((ex){  
    throw ex;  
  });  
}
```

Figura 23. Obtener temperatura del sensor

Elaborado por: El Investigador

La figura 24 muestra la conexión que se establece a la api alojada en el host de railway, se establece una cadena de conexión de tipo estática.

```
class enviroment {  
  static String host="https://fastapi-production-7869.up.railway.app/";  
}
```

Figura 24. Conexión a la API en el host

Elaborado por: El Investigador

Pages

La figura 25 muestra la clase preguntas que van en el aplicativo con sus respectivas opciones que retorna la colección de datos obtenidos en la predicción.

```
class PreguntasPage extends StatefulWidget {
  PreguntasPage({Key? key}) : super(key: key);

  final List<Map<String, dynamic>> _preguntas = [
    {
      'pregunta': 'Temperatura',
      'tipo': 'label',
      'opciones': '36.0',
    },
    {
      'pregunta': '¿Cuál es su Género?',
      'tipo': 'select',
      'opciones': ['Masculino', 'Femenino'],
    },
    {
      'pregunta': '¿Tiene secreción nasal?',
      'tipo': 'select',
      'opciones': ['Si', 'No'],
    },
    {
      'pregunta': '¿Tiene congestión nasal?',
      'tipo': 'select',
      'opciones': ['Si', 'No'],
    },
    {
      'pregunta': '¿Le duele la garganta?',
      'tipo': 'select',
      'opciones': ['Leve', 'Moderado', 'Fuerte', 'No'],
    },
    {
      'pregunta': '¿Tiene lagrimeo?',
      'tipo': 'select',
      'opciones': ['Si', 'No'],
    },
  ],
}
```

Figura 25. Preguntas en Flutter

Elaborado por: El Investigador

En la figura 26 se muestra la función la cual limpia los campos y les inicializa con los valores iniciales

```
void initState() {  
  super.initState();  
  respuestas = List.generate(widget._preguntas.length, (index) => null);  
  GetTemperatura();  
  Timer(const Duration(seconds: 10), () {  
    GetTemperatura();  
  }); // Timer  
}
```

Figura 26. Inicializar los valores al aperturar el app

Elaborado por: El Investigador

IA

En Visual Studio Code se ha codificado la IA que se uso la librería de sklearn para la creación de los modelos de selección de decisiones.

Se ha estructurado en dos archivos de Python los cuales esta clasificado que en el primer archivo denominado F1 se realice la creación de los modelos a ocuparse y a guardarse como archivos independientes ya entrenados.

Código de creación de la IA

Dataset

El dataset utilizado contiene información sobre los diferentes síntomas y signos de las enfermedades respiratorias propuestas, tales como fiebre, dolor de cabeza, dolor de estómago. Cada fila en la base de datos representa a un paciente que ha sido atendido y diagnosticado en el Hospital Básico San Cayetano y cada columna representa a un síntoma o signo de cada enfermedad. Este dataset ha sido proporcionado por el centro medico en mención y se utiliza para registrar las enfermedades respiratorias de cada paciente junto con los síntomas que las caracterizan a cada una, esta información se usó para entrenar el

modelo de aprendizaje automático (Random Forest) para que a partir de ello se pueda clasificar a los pacientes acorde a sus distintas enfermedades basándose en los síntomas que presentan (ver Anexo B).

Tabla 6. Descripción del dataset

Elaborado por: El Investigador

Variable	Tipo de dato	Valores
Temperatura	String	Si, No
Género	String	Masculino, Femenino
Secreción nasal	String	Si, No
Congestión nasal	String	Si, No
Dolor de garganta	String	Leve, Moderado, Fuerte, No
Lagrimo	String	Si, No
Tos	String	Seca, Con flema, No
Estornudos	String	Si, No
Sensación de ahogo	String	Si, No
Fiebre	String	Si, No
Dolor articular	String	Leve, Moderado, Fuerte, No
Malestar general	String	Si, No
Dolor de cabeza	String	Leve, Moderado, Fuerte, No
Picazón nasal	String	Si, No
Hinchazón de ojos	String	Si, No
Ronquidos	String	Si, No
Dolor muscular	String	Si, No
Pérdida de voz	String	Si, No
Diarrea	String	Si, No
Nauseas	String	Si, No
Dolor estomacal	String	Si, No
Dolor de pecho	String	Si, No
Pérdida de apetito	String	Si, No

Escalofríos	String	Si, No
Diagnóstico	String	COVID-19, Rinitis, Resfriado

Predictores

Son las variables de entrada que se utilizan para predecir o estimar el valor de una variable objetivo o de respuesta. El dataset utilizado en este proyecto de investigación contiene los siguientes predictores: género, secreción nasal, congestión nasal, dolor garganta, lagrimeo, tos, estornudos, sensación de ahogo, fiebre, dolor articular, malestar general, dolor de cabeza, picazón nasal, hinchazón de ojos, ronquidos, dolor muscular, pérdida de voz, dolor de ojos, diarrea, nauseas, dolor de estómago, dolor pecho, perdida de apetito y escalofríos (ver Anexo B).

Definición de las clases

Las clases son etiquetas que se asignan a un conjunto de datos para que el modelo pueda aprender a reconocer y clasificar nuevos datos dentro de esa misma categoría. En este caso, las clases utilizadas en este proyecto son las que se muestran en la figura 20.

```

40 # Valida clases en el dataset
41 classes = datos['Diagnostico'].unique().tolist()
42 print(f'Clases: {classes}')
43
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python +
PS D:\Documentos Importantes\Tesis\ia> & C:/Users/sebas/AppData/Local/Programs/Python/Python311/python.exe "d:/Documentos Importantes/Tesis/ia/F1-entrenamiento.py"
Clases: ['RESFRIADO', 'COVID19', 'RINITIS']

```

Figura 27. Definición de clases usadas en el dataset

Elaborado por: El Investigador

Entrenamiento del modelo

La figura 28 muestra como se transforman las opciones de texto a numéricas tomando todas las que se tienen en la base de datos para el entrenamiento del algoritmo, por lo que se procedió a crear una función tomando todas las opciones de las preguntas planteadas y asignándoles un número en su reemplazo.

```
#-----Transformar datos a numericos-----  
def transformarDatos(datos):  
    datos=pd.DataFrame([datos],columns=columnas.columns)  
    print(datos)  
  
    datos[datos == 'si'] = 1  
    datos[datos == 'no'] = 0  
  
    datos[datos == 'femenino'] = 1  
    datos[datos == 'masculino'] = 0  
  
    datos[datos == 'leve'] = 1  
    datos[datos == 'moderado'] = 2  
    datos[datos == 'fuerte'] = 3  
  
    datos[datos == 'seca'] = 1  
    datos[datos == 'con flema'] = 0  
    print(datos)  
  
    return datos
```

Figura 28. Función Transformar los datos

Elaborado por: El Investigador

La figura 29 indica como se secciona la información de la base de datos tomando todas las columnas que corresponden a los síntomas respiratorios como variables independientes y el diagnóstico o resultado se toma como variable dependiente para luego generar el entrenamiento.

```
#-----Separar Variable dependiente y independiente-----  
x=datos.drop(columns='Diagnostico')  
y = datos['Diagnostico']
```

Figura 29. Separación de datos

Elaborado por: El Investigador

En la figura 30 indica el balanceo y división de los datos segmentando el 70% de datos para entrenamiento mientras que el 30% para prueba o validación, el parámetro stratify permite balancear a los datos de prueba y entrenamiento de la variable y, la variable random_state permite que los datos de prueba y entrenamiento sean los mismos.

```
from sklearn.model_selection import train_test_split
xEntrenamiento, xPrueba, yEntrenamiento, yPrueba = train_test_split(x,y,test_size=0.3,stratify=y,random_state=44)
```

Figura 30. Balanceo y división de entrenamiento

Elaborado por: El Investigador

En la figura 31 indica como se entrena el modelo de Random Forest, tomando como parámetro n_estimators que representa el numero de arboles que se ocupan en el modelo, posteriormente se agrupa los datos pre entrenados de la variable dependiente e independiente.

```
# RANDOM FOREST-----
rf = ensemble.RandomForestClassifier(n_estimators=20)
rf.fit(xEntrenamiento, yEntrenamiento)
```

Figura 31. Entrenamiento del modelo

Elaborado por: El Investigador

En la figura 32 indica como se crea la matriz de confusión del modelo, tomando en cuenta como se genera la predicción con los síntomas de las enfermedades respiratorias propuestas.

```
|
print("-----RANDOM FOREST-----")
y_pred = rf.predict(xPrueba)
print("-----")
```

Figura 32. Creación de la Matriz de Confusión

Elaborado por: El Investigador

En la figura 33 se muestra la matriz de confusión que representa los resultados de una clasificación de tres clases, tiene tres filas y tres columnas, donde las filas representan las clases reales y las columnas representan las clases predichas por el modelo. Los números en la diagonal principal representan el número de muestras que fueron correctamente clasificadas para cada clase.

- En la primera fila, hay 818 muestras que pertenecen a la primera clase que fueron correctamente clasificadas.
- En la segunda fila, hay 790 muestras que pertenecen a la segunda clase que fueron correctamente clasificadas.
- En la tercera fila, hay 545 muestras que pertenecen a la tercera clase que fueron correctamente clasificadas.

Los números fuera de la diagonal principal representan las muestras que fueron incorrectamente clasificadas.

- En la primera fila, hay 154 muestras que pertenecen a la primera clase que fueron clasificadas como la segunda clase.
- En la segunda fila, hay 97 muestras que pertenecen a la segunda clase que fueron clasificadas como la primera clase.
- En este caso, no hay muestras que fueron clasificadas incorrectamente en la tercera clase.

```

-----RANDOM FOREST-----
[[818 154  0]
 [ 97 790  0]
 [  0  0 545]]
-----

```

	precision	recall	f1-score	support
COVID19	0.89	0.84	0.87	972
RESFRIADO	0.84	0.89	0.86	887
RINITIS	1.00	1.00	1.00	545
accuracy			0.90	2404
macro avg	0.91	0.91	0.91	2404
weighted avg	0.90	0.90	0.90	2404

Figura 33. Procesamiento de la Matriz de Confusión

Elaborado por: El Investigador

Validación de entrenamiento y prueba

En la figura 34 indica como se obtiene la validación cruzada del modelo para validar el resultado del entrenamiento de los datos y comprobar que no existe overfitting en el modelo, con ello se calcula la precisión del entrenamiento y la precisión de prueba.

```
print("Validacion Cruzada - RANDOM FOREST")
print("Precision de entrenamiento: " + f'{cross_val_score(rf, xEntrenamiento, yEntrenamiento,
print("Precision de prueba: " + f'{cross_val_score(rf, xPrueba, yPrueba, cv=5).mean():.3f}%')
```

Figura 34. Validación Cruzada

Elaborado por: El Investigador

En la figura 35 se muestra el resultado de la validación cruzada, la precisión de entrenamiento sirve para detectar si el modelo esta sobreentrenado o existe la falta del mismo. La precisión de prueba es la validación en base a datos reales como se comporta el modelo entrenado.

```
Validacion Cruzada - RANDOM FOREST
Precision de entrenamiento: 0.903%
Precision de prueba: 0.892%
```

Figura 35. Resultado de la Validación Cruzada

Elaborado por: El Investigador

En la figura 36 indica como se crea el archivo pkl para guardar el modelo en un archivo separado, por lo cual se genera la estructura del archivo tomando como parámetro la ruta de la carpeta en la cual se esta trabajando actualmente mas el nombre y extensión que va a tener dicho archivo.

```
print("RANDOM FOREST")
archivo_joblib = pathDeTrabajo+pathModelos+"Random-Forest.pkl"
joblib.dump(rf,archivo_joblib)
```

Figura 36. Guardar el modelo en un archivo pkl

Elaborado por: El Investigador

Predicción

En la figura 37 indica como se carga el modelo y se crea la función predecir, partiendo de una comparativa que se realiza por medio de una condición para saber si el archivo del modelo ya ha sido creado para cargar dicho modelo y que genere la predicción con la función que se ha creado denominado predecir el cual devuelve los datos ya entrenados con los que ingrese el usuario por medio del asistente médico virtual.

```
#Cargando modelo RF
if os.path.isfile(pathDeTrabajo+pathModelos+'Random-Forest.pkl'): # Si existe
    modelRF = joblib.load(pathDeTrabajo+pathModelos+"Random-Forest.pkl")
else:
    print('El archivo: '+pathDeTrabajo+pathModelos+'Random-Forest.pkl no existe')
    exit

# Predice usando modelo
def predecir(datos):
    # Random Forest
    return modelRF.predict(datos[0:1])
```

Figura 37. Carga del modelo Random Forest y predicción

Elaborado por: El Investigador

En la figura 38 se indica como se inicializa la librería fastapi, se crea el modelo denominado síntomas para el json del request de la petición de la api y luego establece los dos endpoints del api, siendo el primero el post en el URL predecir y el segundo el get en la raíz de la ruta.

```
#Armar Api
app = FastAPI()

class Sintomas(BaseModel):
    sintomas: list = []

@app.post("/predecir", status_code=201)
async def apiPrecedir(sintomas: Sintomas):
    datos = transformarDatos(sintomas.sintomas)
    diagnostico = predecir(datos)[0]
    guardarRegistro(datos, diagnostico)
    resp = {'estado':{'codigo':0, 'mensaje':''}, 'payload':diagnostico}
    return resp

@app.get("/", status_code=201)
async def getStatus():
    resp = {'estado':{'codigo':0, 'mensaje':''}, 'payload':'OK'}
    return resp
```

Figura 38. Creación de la api

Elaborado por: El Investigador

Railway

Para el deploy de la API se tiene que crear un proyecto con los requerimientos necesarios del proyecto a ejecutar, y una vez establecido la estructura se enlaza el programa, el cual puede ser de manera manual el código o usando una plantilla y compilar.

Como se muestra en la figura se crea un proyecto en Railway de tipo FastAPI

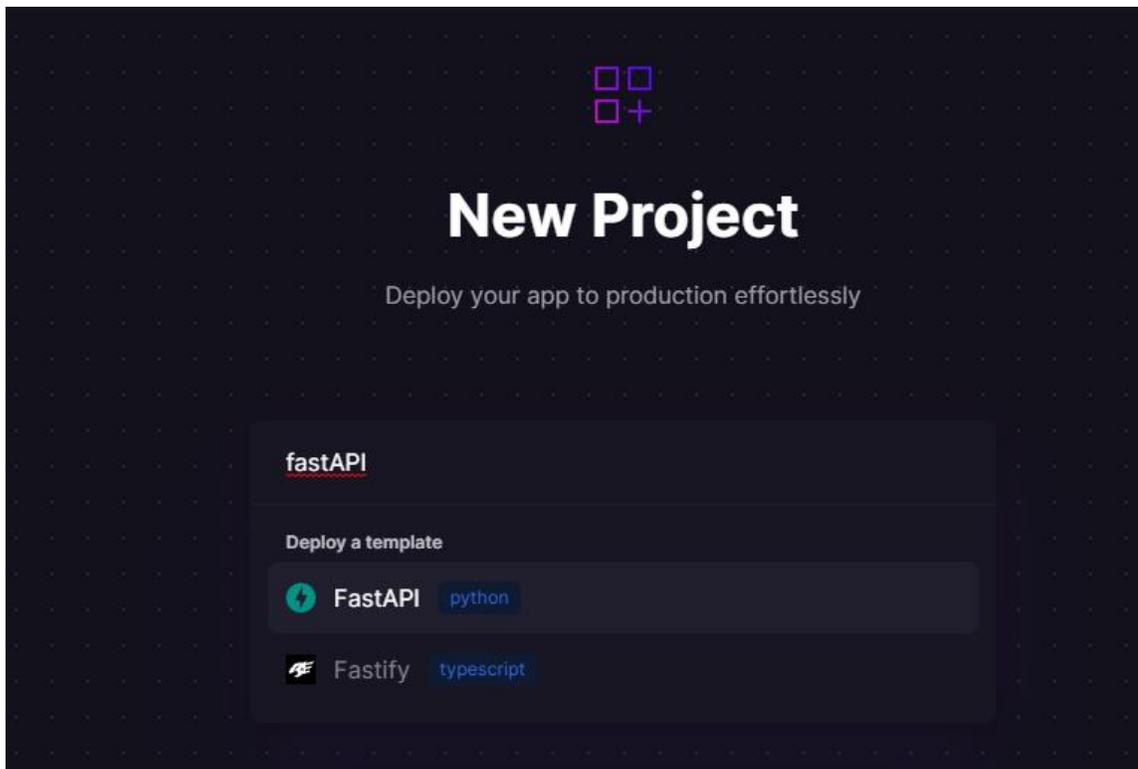


Figura 39. Creación de Proyecto Railway

Elaborado por: El Investigador

En la figura 40 se indica la estructura del proyecto que se obtiene al seleccionar la plantilla de FastAPI, se agrega los archivos referentes al modelo de Random Forest y el archivo de conexión a Firebase.

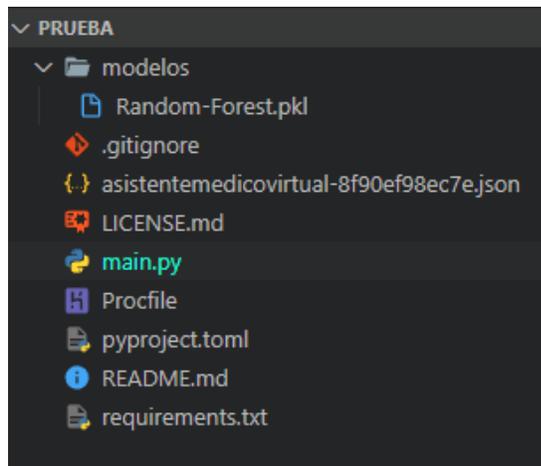


Figura 40. Estructura proyecto FastAPI - Railway

Elaborado por: El Investigador

En la figura 41 se muestra las librerías usadas en el proyecto en formato para compilación de Railway

```
requirements.txt
6 idna==3.3
7 pydantic==1.9.2
8 python-dotenv==0.20.0
9 PyYAML==6.0
10 sniffio==1.2.0
11 starlette==0.19.1
12 typing_extensions==4.3.0
13 uvicorn==0.18.3
14 uvloop==0.16.0
15 watchfiles==0.16.1
16 websockets==10.3
17 pandas==1.4.3
18 joblib==1.2.0
19 firebase_admin==6.0.1
20 scikit_learn==1.1.2
21
```

Figura 41. Librerías y sus versiones en formato Railway

Elaborado por: El Investigador

En la figura 42 se muestra la cadena de conexión a la Base de Datos establecida en Firebase al proyecto Asistente Médico Virtual.

```

import firebase_admin
from firebase_admin import credentials
from firebase_admin import firestore

cred = credentials.Certificate(pathDeTrabajo+"asistentemedicovirtual-8f90ef98ec7e.json")
firebase_admin.initialize_app(cred)
db = firestore.client() # connecting to firestore

```

Figura 42. Cadena de conexión a Firebase

Elaborado por: El Investigador

En la figura 43 se indica la función que permite guardar la colección de datos en la Base de Datos

```

def guardarRegistro(datos,diagnostico):
    #Guardar datos en db
    datos['diagnostico']=diagnostico
    json = datos.to_dict(orient='index')[0]
    print(json)
    doc_ref = db.collection(u'prime_location').document()
    docId = doc_ref.id #generates id
    print('ID: ', docId) #print the generatd id
    collection = db.collection('sintomas') # create collection
    res = collection.document(docId).set( json )
    print(res)

```

Figura 43. Función guardar registro

Elaborado por: El Investigador

En la figura 44 se muestra la función la cual obtiene la ultima temperatura registrada en la Base de Datos

```
def obtenerUltimaTemperatura():  
    collection = db.collection('temperatura') # create collection  
    res = collection.document('T').get().to_dict()  
    print(res)  
    return res['temp']
```

Figura 44. Función obtener temperatura

Elaborado por: El Investigador

En la figura 45 se muestra la ejecución de la API en Railway y los cambios realizados mediante los commits emitidos en GitHub.

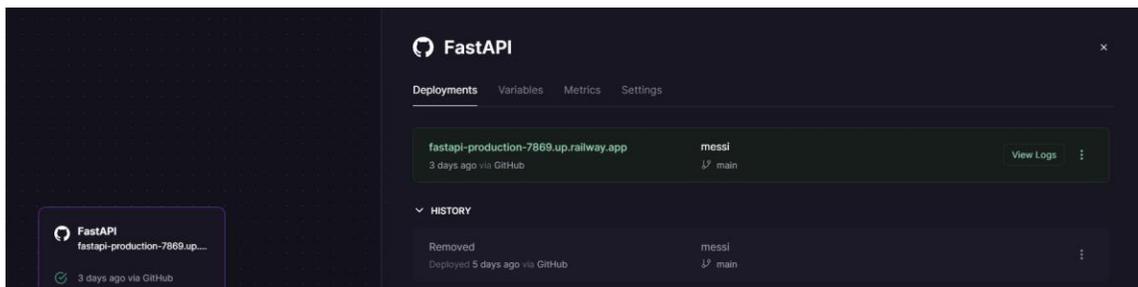


Figura 45. API ejecutándose en Railway

Elaborado por: El Investigador

3.2.3. Fase de Estabilización

En esta fase va principalmente los cambios que se han realizado en el desarrollo del software.

Al momento de empezar con el desarrollo y el diseño del aplicativo y la ejecución de la API se han realizado cambios significativos como:

- Cambio de tipo de ejecución de la API de manera local con uvicorn a alojarlo en Railway.

- Implementación del campo temperatura el cual es tomando mediante ultimo dato subido con anterioridad a Firebase además de integrarse al Proyecto de Investigación denominado “Sistema electrónico de monitoreo de bioseñales y alertas de distanciamiento social para la prevención de SARS-COV2 aplicando Inteligencia Artificial”
- Mejora en el diseño del aplicativo móvil al implementar un fondo de pantalla mas amigable con el usuario y un rediseño de los mensajes y preguntas a mostrarse.

3.2.4. Fase de Pruebas

En esta fase se verifica el funcionamiento de manera correcta del aplicativo móvil y en caso de existir algún inconveniente o error dar solución al mismo con la finalidad de solventar y que el aplicativo quede optimo para el uso y permita realizar diagnósticos a pacientes.

Pruebas de las Funcionalidades

Pantalla de Inicio

Al ingresar al aplicativo el paciente observa en la pantalla de inicio el mensaje de bienvenida el explica cual el funcionamiento del Asistente Médico Virtual, en base a las opciones que seleccione de los síntomas presentados y de esta manera obtener un diagnostico y con ello acuda a un centro medico para una revisión y garantice el cuidado de la salud del paciente.

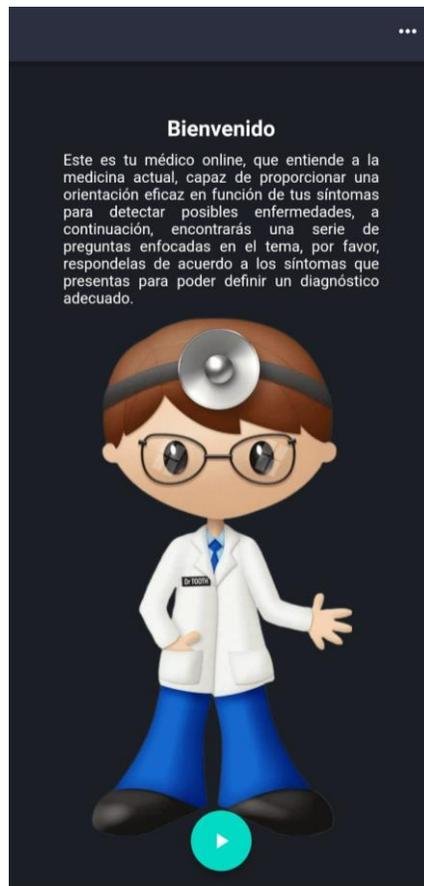


Figura 46. Pantalla de Bienvenida

Elaborado por: El Investigador

Pantalla de la Primera Pregunta

El paciente selecciona el valor de la temperatura que previamente se le ha tomado por medio del sensor del proyecto de Investigación (ver Anexo C) el cual es un valor único obtenido en la Base de Datos alojada en Firebase, programado con el aplicativo para que el último valor guardado en la Colección Temperatura se obtenga y se refleje en el Asistente Médico Virtual.



Figura 47. Obtención de la temperatura desde el dispositivo

Elaborado por: El Investigador

Pantalla del Diagnóstico

El paciente una vez haya finalizado su test en base a los síntomas y signos registrados va a obtener un diagnóstico y un mensaje informativo el cual guía al usuario a precautelar su salud y acudir a una entidad médica cercana.

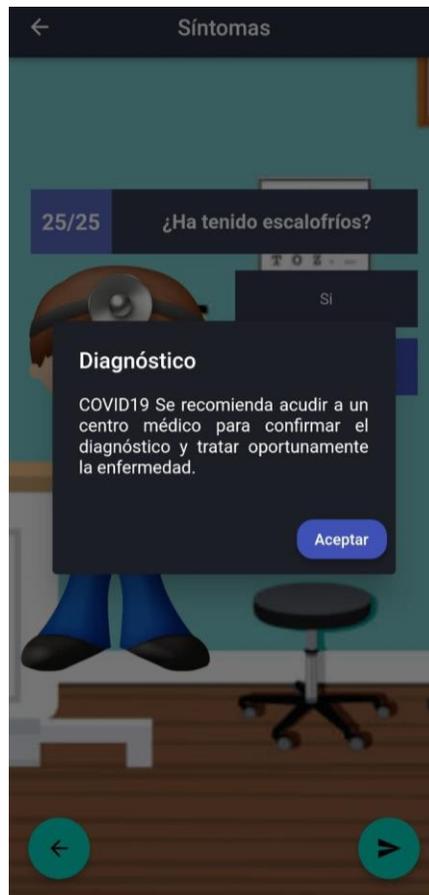


Figura 48. Mensaje de diagnostico en la pantalla final

Elaborado por: El Investigador

Pruebas de las Aceptación

En base a las pruebas realizadas sobre el funcionamiento del Asistente Médico Virtual, se obtuvieron lo siguiente:

Tabla 7. Prueba de aceptación

Elaborado por: El Investigador

Prueba de aceptación	Número: 1
Nombre: Diagnóstico	
Descripción: La funcionalidad permite al paciente obtener un resultado en base a sus síntomas.	
Condiciones de ejecución: El paciente como primera instancia debe seleccionar la temperatura y posteriormente cada signo y síntoma que tenga, luego debe enviar sus datos para que se genere la comparativa con los datos entrenados.	
Resultado esperado: Al paciente se le mostrará el diagnóstico acompañado de un mensaje para que acuda a un centro médico y reciba atención inmediata.	
Resultado de prueba: Prueba satisfactoria.	

CAPITULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- Los información obtenida de la Base de Datos del Hospital Básico “San Cayetano” de los cuales el 70% fue utilizado para el entrenamiento del algoritmo y el 30% para la validación del mismo, esto sirvió para entrenar el algoritmo de IA y crear la API, con ello se pudo generar la predicción de los datos ingresados por los pacientes y así garantizar la funcionalidad del aplicativo de tipo Asistente Médico Virtual.
- Se determinó que el mejor método de aprendizaje para clasificar en base a características o variables predictoras y que también sea útil al tener valores perdidos o valores atípicos es el Random Forest ya que su ejecución consiste en la identificación de valores únicos que permitan la toma decisiones.
- El análisis comparativo entre frameworks más usados en proyectos similares a este permitió establecer el uso de Flutter para la ejecución del presente proyecto, ya que el mismo permitió realizarlo de una manera más factible y ágil debido a su fácil uso.
- Se realizó la implementación del asistente medico virtual usando el algoritmo Random Forest debido a su capacidad para manejar gran cantidad de datos, este algoritmo tiene la ventaja de estar conformado por varios árboles de decisión donde cada uno de ellos posee distintas características y conjunto de datos, por esa razón, al momento de tomar una decisión el algoritmo promedia la predicción de todos los árboles individuales en el bosque dando como resultado una predicción final más precisa y estable.
- Se realizó la integración del sistema electrónico de monitoreo con el asistente virtual a través de la conexión de ambos proyectos a la base de datos Firebase, se enlazó el valor de la temperatura en la app para usarlo como una de las variables o síntomas del paciente y posterior a ello enviar en el conjunto de datos para obtener un resultado o diagnóstico.

4.2. Recomendaciones

- Se recomienda promover el diagnóstico a través de inteligencias artificiales de las distintas enfermedades respiratorias, de esta manera se puede lograr que los pacientes acudan a especialistas, precautelando la salud de cada persona y evitando la saturación del sistema de salud.
- Se recomienda capacitar a los estudiantes para que ocupen herramientas virtuales para la detección de enfermedades tales como asistentes médicos y así poder llevar un control mas riguroso del cuidado de la salud.
- Se recomienda usar Flutter ya que brinda muchos beneficios y es fácil uso e implementación y existe documentación para cualquier tipo de desarrollo de aplicaciones móviles.
- Se recomienda el uso de Railway como host ya que en la investigación realizada no tiene competidor directo, es el mas accesible en cuestión a costos y tiene un entorno muy amigable con el usuario que permite guiarse fácilmente, además cuenta con plantillas para el diferente tipo de lenguaje que se quiera alojar en el host.

BIBLIOGRAFÍA

- [1] D. N. A. R. K. Z. H. F. T. M. de O. M. B. N. Marciniuk, “Las enfermedades respiratorias en el mundo”.
- [2] J.-C. Vázquez-García, J. Salas-Hernández, R. Pérez Padilla, and M. Montes de Oca, “Salud respiratoria en América Latina: número de especialistas y formación de recursos humanos,” *Arch. Bronconeumol.*, vol. 50, no. 1, pp. 34–39, Jan. 2014, doi: 10.1016/J.ARBRES.2013.07.011.
- [3] J. LLC, “JustAnswer,” 2023, [Online]. Available: <https://www.justanswer.com/>
- [4] G. S. Rosero, “PillBot, tu asistente médico virtual,” 2023, [Online]. Available: <http://bancodeideas.gob.ec/proyecto/view?data=aWQ9OTk0OA%3D%3D>
- [5] J. Garcia, “Desarrollo de un chatbot como apoyo a la experiencia del cliente en un hotel en Colombia,” 2022, [Online]. Available: <https://repositorio.uniandes.edu.co/bitstream/handle/1992/55148/26483.pdf?sequence=1&isAllowed=y>
- [6] R. A. Lindao Alfonso, Génesis Adriana & Castañeda Auquilla, “Desarrollo De Un Asistente Virtual (Chatbot) Para Facilitar El Acceso a La Información Requerida Por Los Estudiantes De La Facultad De Ciencias Matemáticas Y Físicas De La Universidad De Guayaquil,” *Univ. Guayaquil. Fac. Ciencias Matemáticas y Físicas. Carrera Ing. en Sist. Comput.*, p. 195, 2020, [Online]. Available: <http://repositorio.ug.edu.ec/handle/redug/49579>
- [7] M. Montoya and J. Sánchez, “Desarrollo De Un Prototipo De Aplicación Móvil Utilizando Sdk Flutter Y Lenguaje De Código Abierto Dart Para Promover Actividades Deportivas En Guayaquil,” pp. 1–168, 2020, [Online]. Available: <http://repositorio.ug.edu.ec/handle/redug/49571>
- [8] A. I. Humberto Martínez Urrea, Martínez Humberto, Alzate Diego, Ríos Marlon and C. V. Archila Johanna, “Factores de riesgo a enfermedades respiratorias,” *Rev. Mex. Pediatr.*, vol. 76, pp. 252–255, 2009.

- [9] S. Pérez, “SISTEMA ELECTRÓNICO DE MONITOREO DE SIGNOS VITALES Y ALERTAS DE DISTANCIAMIENTO SOCIAL PARA LA PREVENCIÓN DE ENFERMEDADES RESPIRATORIAS.” <https://repositorio.uta.edu.ec/bitstream/123456789/33605/1/t1870ec.pdf> (accessed Jul. 12, 2022).
- [10] Organización Mundial de la Salud, “Prevención y control de las infecciones respiratorias agudas con tendencia epidémica y pandémica durante la atención sanitaria”, Accessed: Jul. 12, 2022. [Online]. Available: www.who.int
- [11] C. Femenia, Jose; Mollar, “GUÍA BÁSICA ENFERMEDADES RESPIRATORIAS,” vol. 15, no. 2, pp. 1–23, 2016.
- [12] M. enciclopedia Médica, “Enfermedad pulmonar,” 2022. <https://medlineplus.gov/spanish/ency/article/000066.htm> (accessed Mar. 05, 2023).
- [13] I. N. de S. de USA, “Definición de enfermedad respiratoria - Diccionario de cáncer del NCI - NCI.” <https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/enfermedad-respiratoria> (accessed Jul. 12, 2022).
- [14] Teijin Esteve, “Patologías respiratorias .” <https://www.esteveteijin.com/patologias-respiratorias/> (accessed Jul. 12, 2022).
- [15] C. para el C. y la P. de Enfermedades, “Gripe Común”, [Online]. Available: <https://espanol.cdc.gov/flu/index.htm>
- [16] M. Salazar, J. Barochiner, W. Espeche, and I. Ennis, “COVID-19, hipertensión y enfermedad cardiovascular,” *Hipertens. y Riesgo Vasc.*, vol. 37, no. 4, pp. 176–180, Oct. 2020, doi: 10.1016/J.HIPERT.2020.06.003.
- [17] MedlinePlus, “Rinitis alérgica: MedlinePlus enciclopedia médica.” <https://medlineplus.gov/spanish/ency/article/000813.htm> (accessed Aug. 26, 2022).

- [18] J. L. Viejo Bañuelos, “Infecciones agudas de la vía aérea superior,” in *Neumología Clínica*, Elsevier, 2010, pp. 271–278. doi: 10.1016/b978-84-8086-298-1.50034-2.
- [19] J. Hernández, “Análisis y Desarrollo Web,” *Books.Google.Com*, 2018. <https://books.google.es/books?id=nYDVBQAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false> (accessed Jun. 17, 2022).
- [20] P. Kebin, “Diseño de un videojuego para el rescate cultural identitario en adolescentes Ecuatorianos,” 2019.
- [21] Oracle México, “¿Qué es la inteligencia artificial (IA)?” <https://www.oracle.com/mx/artificial-intelligence/what-is-ai/> (accessed Jul. 13, 2022).
- [22] Grupo Atico34, “Inteligencia artificial: Definición, tipos y aplicaciones.” https://protecciondatos-lopd.com/empresas/inteligencia-artificial/#Que_es_la_inteligencia_artificial (accessed Jul. 13, 2022).
- [23] Á. Menacho Rodríguez, “EventYou: Aplicación móvil con Flutter y FlutterFire,” *Univ. Sevilla*, pp. 1–67, 2022, [Online]. Available: <https://idus.us.es/bitstream/handle/11441/102882/TFG-3024-MENACHO RODRIGUEZ.pdf?sequence=1&isAllowed=y>
- [24] “Firebase.” <https://firebase.google.com/?hl=es> (accessed Jan. 08, 2023).
- [25] E. Agüero, “Firebase en el desarrollo de aplicaciones móviles,” *Univ. Politécnica Sinaloa*, p. 70, 2018, [Online]. Available: <http://repositorio.upsin.edu.mx/formatos/TesinaEdgarUlisesAgueroAguiar6855.pdf>
- [26] “Concepto Railway.” <https://docs.railway.app/getting-started> (accessed Jan. 08, 2023).
- [27] U. A. de Chile, “Validación Por Expertos,” *Validación Por Expert.*, pp. 1–22, 2019, [Online]. Available: https://www.unach.cl/wp-content/uploads/2018/06/INSTRUMENTOS_Validacion_expertos_cuestionario.d

OCX

ANEXOS

ANEXO A

Certificado del Hospital Básico “San Cayetano”



Ambato, 20 de octubre del 2022

CERTIFICADO

Certifico que el Sr. **DAVID SEBASTIAN GONZALEZ ALBAN** con CI: **1804776696** estudiante de la UNIVERSIDAD TECNICA DE AMBATO, de la facultad de INGENIERA EN SISTEMAS, ELECTRONICA E INDUSTRIAL, de la carrera de INGENIERIA EN TECNOLOGIAS DE LA INFORMACION, solicitó los datos de las enfermedades: resfriado común, rinitis alérgica y covid-19.

Datos que servirán para su proyecto de titulación con el tema “Asistente médico virtual orientado al diagnóstico de enfermedades respiratorias aplicando inteligencia artificial.”

Es todo cuanto puedo certificar para los fines consiguientes, exceptuando trámites legales.

Dr. Franklin Nuñez
CODIGO 1803190717



DR. FRANKLIN NUÑEZ
DIRECTOR MÉDICO
CI: 1803190717



Dirección: Calle Daquilema y Roca Inca (Esquina)
Teléfono: 2 84 09 08 Celular: 0939905862
E-mail. bio_diagnostico_lab@yahoo.es
Ambato – Ecuador

ANEXO B

Base de Datos del Hospital Básico “San Cayetano”

Paciente	Genero	Secrecion	Congestio	Dolor garg	Lagrimeo	Tos	Estornudc	Sensacion	Fiebre	dolor artic	Malestar	g	Dolor cabi	Picazon n	Hinchazor	Ronquido	Dolor mus	Perdida v	Dolor ojos	Diarrea	Nauseas	Dolor barr	Dolor pecl	Perdida
1	femenino			leve		seca																		
2	femenino	si					si						leve						si					
3	femenino	si	moderado						si	moderado														
4	femenino	si							no															
5	masculino	si	fuerte			seca																		
6	femenino	si							si															
7	femenino	si	leve																					
8	masculino	si							si			leve												
9	femenino	si	leve			seca				si		leve												
10	masculino	si	moderado						no															
11	masculino	si				con flema																		
12	masculino	si											moderado											
13	femenino	si				seca																		
14	femenino	si							si	si														
15	femenino	si	moderado																					
16	femenino	si					si			si		leve												
17	masculino	si				con flema			si															
18	femenino					con flema			si	leve	si													
19	masculino	si	moderado			con flema					si													
20	femenino					con flema				moderado														
21	femenino	si	leve			seca																		
22	femenino	si							si															
23	femenino	si				con flema			si															
24	femenino	si	leve																					
25	femenino	si	moderado									leve												
26	femenino	si	leve				si																	
27	masculino	si				seca																		
28	femenino		moderado							moderado	si											si		
29	masculino	si				con flema																		
30	femenino								si			fuerte												
31	masculino					con flema					si							si						
32	masculino	si							si		si													
33	masculino	si	leve			seca																		

ANEXO C

Pulsera Electrónica vinculada al proyecto de Investigación



ANEXO D

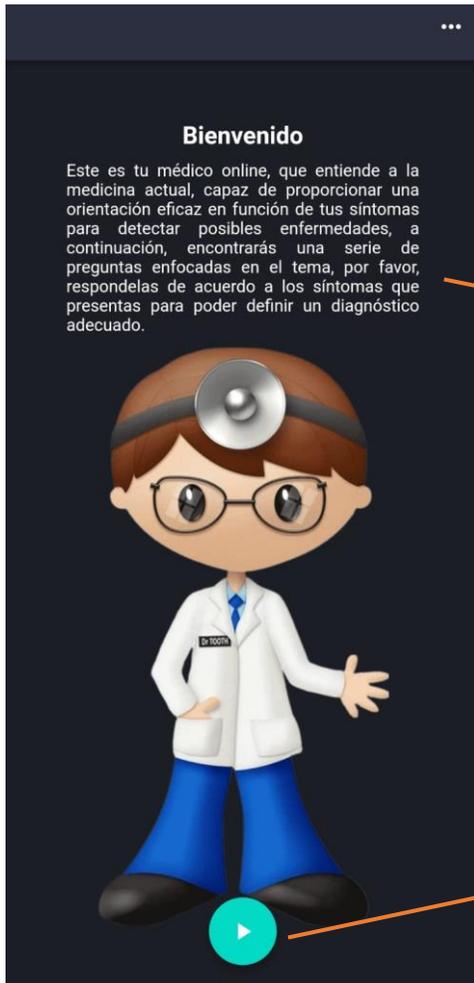
Manual de Usuario

“ASISTENTE MEDICO VIRTUAL ORIENTADO AL DIAGNÓSTICO DE ENFERMEDADES RESPIRATORIAS APLICANDO INTELIGENCIA ARTIFICIAL”

MANUAL DE USUARIO

El presente manual de usuario se realizó con el objetivo de guiar al paciente acerca del uso de la funcionalidad de la Aplicación Móvil desarrollada sobre el diagnóstico de enfermedades respiratorias aplicando inteligencia artificial.

Pantalla de Inicio



El paciente puede entender el propósito y la funcionalidad del asistente médico virtual mediante el mensaje de bienvenida.

Para ingresar al aplicativo se deber dar clic sobre el botón e ingresa a las preguntas.

Pantalla de Primera Pregunta

El botón de flecha en la parte superior permite al usuario regresar a la pantalla del mensaje de bienvenida.



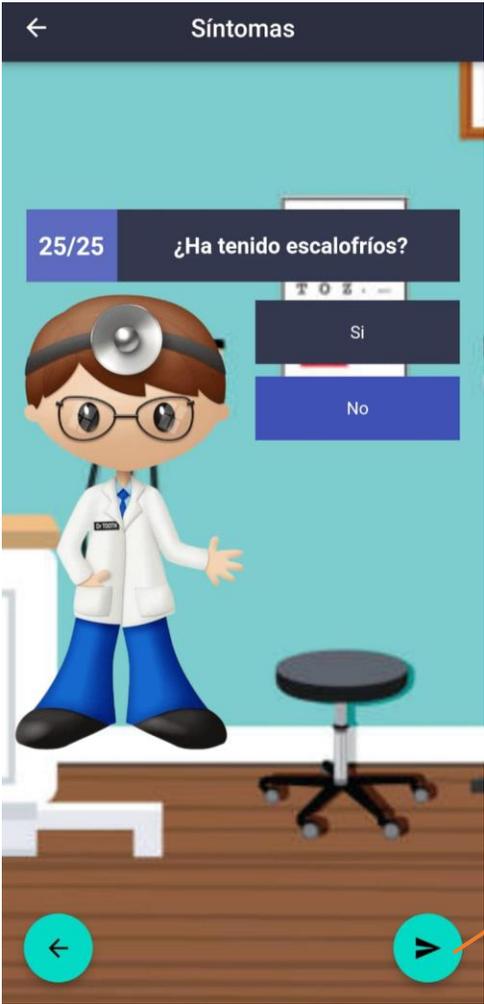
El paciente puede visualizar la temperatura tomada previamente con el sensor y se muestra como la primera pregunta del asistente médico virtual.

Se tiene un botón de una flecha a la derecha para avanzar en la siguiente pregunta.

Pantalla de Preguntas



Pantalla de Última pregunta



El botón envía los datos y se obtiene el diagnóstico con base en los síntomas registrados.

Pantalla de Diagnostico



Se muestra una ventana flotante la cual se evidencia el diagnóstico y un mensaje informativo para precautelar la salud del paciente.