



# COVID-X: Sistema de ayuda en la toma de decisiones en el diagnóstico médico para la enfermedad COVID-19.

## **Autores:**

Alejo Ezequiel Aquili

Santiago Bassani

Francisco Javier Sanguinetti Arena

## **Tutores:**

Dra. Alicia Laura Mon

Lic. Sergio Gustavo Herrera

PROYECTO FINAL INGENIERÍA INFORMÁTICA

Instituto Tecnológico de Buenos Aires (ITBA)

Diciembre 2020

## Resumen

En el presente se detallan el diseño y desarrollo de un sistema informático para la ayuda en la toma de decisiones en el diagnóstico médico de la enfermedad COVID-19. Este sistema provee una herramienta que puede detectar neumonía causada por esta enfermedad, manifestada en imágenes rayos-x de tórax. Para lograr esto se implementó un algoritmo basado en redes neuronales convolucionales, y se complementó con otro modelo en base a características clínicas de los pacientes. Basados en los resultados obtenidos en las pruebas experimentales, se muestra la potencial utilidad de estas tecnologías en el proceso de diagnóstico como herramienta complementaria. El sistema desarrollado, llamado COVID-X, no solo incluye el modelo de inteligencia artificial, sino que también una plataforma web que facilita su uso para todo personal médico que lo requiera.

## Abstract

In this work are explained and discussed the design and development of a system for assisting in decision-making in medical diagnosis of the COVID-19 disease. To achieve this, an algorithm based on convolutional neural networks was implemented, and it was complemented with another model based on the clinical characteristics of the patients. Based on the results obtained in the experimental tests, the potential usefulness of these technologies in the diagnostic process as a complementary tool is shown. The system developed, called COVID-X, not only includes the artificial intelligence model, but also a web platform that facilitates its use for all medical personnel who require it.

## Agradecimientos

Los autores muestran su agradecimiento a Juan Li Puma, ingeniero informático cuyo aporte fue indispensable para la realización del proyecto administrando la infraestructura del sistema COVID-X, a Leonardo Sanchez desarrollador que colaboro con el diseño y desarrollo del sitio web de COVID-X, a todo el equipo médico del área de radiología del Hospital Gutierrez por su asistencia, a Ricardo Ruggeri y los especialistas de la clínica de imágenes Leben Salud por su colaboración, al estudio de abogados Bodlegal y al abogado Joaquin Cullen Paunero por el asesoramiento en cuestiones legales, y por último, a los directivos de Intel, Dell y Amazon Web Services por su interés y colaboración en el proyecto.

Se desea extender un especial agradecimiento para Alicia Laura Mon y Sergio Gustavo Herrera quienes llevaron adelante la tutoría del proyecto. Sumar también nuestra inmensa gratitud a Rodrigo Ramele por incentivar y apoyar el desarrollo del proyecto, a Maria Juliana Gambini por su asesoría, y a Santiago Raúl Valles junto a todo el departamento de ingeniería en informática del Instituto Tecnológico de Buenos Aires por su gran apoyo.

# Índice general

<b>Índice general</b>	<b>3</b>
<b>1 Introducción</b>	<b>5</b>
<b>2 Estado del arte</b>	<b>7</b>
2.1 Radiografías de tórax como método de detección de enfermedades pulmonares . . . . .	7
2.2 Complemento a las radiografías: características y comorbilidades	7
2.3 Aprendizaje profundo para la interpretación de radiografías . .	8
<b>3 Extensión y alcance del proyecto</b>	<b>9</b>
<b>4 Producto desarrollado</b>	<b>10</b>
4.1 Requerimientos . . . . .	10
4.1.1 Requisitos funcionales . . . . .	10
4.1.2 Requisitos no funcionales . . . . .	13
4.2 Arquitectura . . . . .	14
4.2.1 Atributos de calidad . . . . .	14
4.2.2 Modelo de vistas . . . . .	16
4.3 Inteligencia artificial . . . . .	36
4.3.1 Marco conceptual . . . . .	36
4.3.2 Arquitectura e implementación . . . . .	38
4.3.3 Conjunto de datos . . . . .	41
4.3.4 Entrenamiento . . . . .	44
4.3.5 Metodología de evaluación de modelos . . . . .	47
4.4 Backend . . . . .	48
4.4.1 DICOMVERTER . . . . .	48
4.4.2 COVID-X-API . . . . .	49
4.4.3 COVID-X-USERS . . . . .	52
4.4.4 INFERENCE-GATEWAY . . . . .	53
4.5 Frontend . . . . .	53
4.5.1 Usuarios de COVID-X . . . . .	54
4.5.2 COVID-X-FRONT . . . . .	54
4.6 Infraestructura . . . . .	56
<b>5 Desarrollo</b>	<b>59</b>

<b>6</b>	<b>Resultados</b>	<b>61</b>
6.1	Evaluación de los modelos de inteligencia artificial . . . . .	61
6.2	Integraciones . . . . .	67
<b>7</b>	<b>Manual de uso</b>	<b>69</b>
7.1	Ingreso y registro de nuevo usuario . . . . .	69
7.2	Carga de datos para entrenamiento . . . . .	72
7.3	Realizar una Inferencia . . . . .	76
7.4	Fundamentos para la investigación . . . . .	79
7.5	COVID-X Blog . . . . .	80
7.6	Recuperar contraseña . . . . .	81
7.7	Salir del sistema . . . . .	84
<b>8</b>	<b>Conclusiones y trabajo futuro</b>	<b>85</b>
	<b>Referencias bibliográficas</b>	<b>86</b>
	<b>Anexos</b>	<b>89</b>
<b>A</b>	<b>Documentación de APIs REST</b>	<b>89</b>
A.1	COVID-X-USERS . . . . .	89
A.2	DICOMVERTER . . . . .	95
A.3	COVID-X-API . . . . .	96
A.4	INFERENCE-GATEWAY . . . . .	105

# 1 Introducción

En el transcurso de la pandemia causada por el COVID-19 durante el año 2020, se han visto efectos devastantes en la salud y bien estar de la población mundial. Por estos motivos, y con fines sociales y solidarios, se decidió llevar a cabo este proyecto.

La enfermedad respiratoria contagiosa y ocasionalmente fatal, COVID-19, se origina por el coronavirus tipo 2 del síndrome respiratorio agudo grave denominado SARS-CoV-2. Los portadores de esta enfermedad pueden manifestar una infección pulmonar y en algunos casos una neumonía viral [24]. Se ha demostrado que la neumonía causada por esta enfermedad es más severa en algunos pacientes y posee una tasa de mortalidad significativa [3], [15].

En este contexto, donde los sistemas sanitarios se vieron colapsados por la propagación exponencial del nuevo virus, es de vital importancia la detección temprana de contagios. Dentro de los mecanismos de detección del virus estandarizados, se encuentra la técnica de la reacción en cadena de la polimerasa con transcriptasa inversa [12], RT-PCR por sus siglas en inglés.

La RT-PCR es una técnica de laboratorio que combina la transcripción inversa de ARN en ADN y la amplificación de un ADN objetivo mediante la reacción en cadena de la polimerasa. Este método es utilizado principalmente para medir la cantidad de un ARN específico. El genoma del virus SARS-CoV-2, compuesto por una sola cadena de ARN, se logró secuenciar permitiendo la rápida implementación de una prueba RT-PCR para detectar el virus en una muestra de material genético.

El diagnóstico por imágenes es un método de gran utilidad para los médicos en la evaluación de la enfermedad. Las radiografías de tórax son ampliamente utilizadas en el ámbito clínico por su rapidez, bajo costo y baja exposición a la radiación en comparación con las tomografías computarizadas (TC) [23]. Estas últimas, proporcionan la anatomía tridimensional que puede facilitar la definición de signos y patrones de patologías pulmonares. Sin embargo, se considera que en general la radiografía es suficiente para diferenciar la neumonía, aunque ocasionalmente resulte compleja su correcta interpretación. Por consiguiente, su análisis preciso es vital para el tratamiento del paciente en situaciones agudas, y para ayudar a identificar la aparición de infecciones de COVID-19. Otro punto a tener en cuenta es que los equipos para realizar TC no son tan frecuentes en el sistema sanitario de la República Argentina.

Es por ello que, el objetivo del proyecto consiste en desarrollar una técnica automatizada y precisa para la detección de neumonía viral utilizando radiografías de tórax como medida provisional para la alerta temprana de

esta enfermedad.

La piedra fundacional para llevar adelante el propósito de COVID-X, es el sistema de inteligencia artificial para la clasificación de imágenes médicas de las radiografías de tórax. Se utilizan redes neuronales convolucionales (CNN por sus siglas en inglés) con el fin de clasificar entre neumonía producto de COVID-19, otras causantes de neumonía y el que no se presenten patologías relacionadas con una infección con el virus SARS-CoV-2.

Adicionalmente, se propone un modelo capaz de complementar la inferencia realizada con información clínica representativa, por lo que se busca recolectar un conjunto de datos que incluya tanto radiografías, así como la información clínica.

Para poder exponer de manera pública este sistema de clasificación, y al mismo tiempo, adquirir los datos, se construye una plataforma web. La misma está inscripta en un ecosistema de servicios que incluyen sistema y control de usuarios, así como también organizaciones médicas o instituciones a las que estos pertenecen. En el diseño de esta plataforma se consideran los factores de precisión, seguridad, escalabilidad, disponibilidad y usabilidad.

## 2 Estado del arte

Con la motivación por la necesidad de iniciar un rápido desarrollo ante la grave situación, se centran los esfuerzos en la investigación del modulo de detección de la enfermedad. Recientes avances en aprendizaje automático profundo (Deep learning) usados en imágenes médicas han demostrado resultados prometedores.

Como punto de partida, se opta por determinar la viabilidad y precisión del uso de algoritmos de clasificación a partir de imágenes clínicas, y si las mismas contienen información significativa. De forma complementaria, se analiza la definición de un conjunto de variables clínicas capaces de complementar la clasificación basada en imágenes.

### 2.1 Radiografías de tórax como método de detección de enfermedades pulmonares

Existe evidencia conformada por numerosos datos clínicos indicando que las imágenes de rayos X se pueden utilizar en el diagnostico de pacientes con COVID-19 [25], [6]. A pesar de que la sensibilidad de la radiografía de tórax se considera baja para la detección temprana de afectaciones pulmonares del virus SARS-CoV-2 [8], es evidente que, en el contexto de una emergencia sanitaria generada por la pandemia, este método de diagnóstico por imágenes puede ser una herramienta útil para el diagnostico en el "día a día" de esta enfermedad.

Jacobi y col. [18] describen patrones comunes de anomalías pulmonares de casos de COVID-19 positivo, en radiografías de tórax. Además, sugieren que la comunidad médica puede confiar en la radiografía de tórax realizada con equipos portátiles por su amplia disponibilidad, y por sus reducidos problemas de control de infección que actualmente limitan la utilización de TC.

### 2.2 Complemento a las radiografías: características y comorbilidades

Para aumentar la confianza en la toma de decisiones sobre la necesidad de aislamiento y tratamiento, sería útil determinar qué características presentadas por los casos sospechosos de infección son los que mejor complementan a las radiografías en la predicción de un diagnóstico positivo. Esto es posible analizando las características del paciente, la trayectoria del caso, las comorbilidades, los síntomas, el diagnóstico y los resultados.

Uno de los primeros estudios publicados [9] mostró un análisis de 262 individuos confirmados como infectados por COVID-19 para determinar sus características clínicas y epidemiológicas en Beijing, China y encontró que las vías de transmisión respiratoria y extra respiratoria pueden explicar la rápida propagación de la enfermedad.

En otro estudio [5], implementan y discuten modelos de aprendizaje automático supervisado para identificar los síntomas clínicos más importantes y significativos que predicen casos de COVID-19 positivos verdaderos. Algunas de las características contempladas incluyen detalles de las personas afectadas, por ejemplo, edad, sexo, y detalles clínicos como la gravedad de la tos y la incidencia de infección pulmonar.

### **2.3 Aprendizaje profundo para la interpretación de radiografías**

Recientemente se ha incrementado el uso de aprendizaje profundo y la recolección de grandes conjuntos de datos en aplicaciones médicas. En algunas de ellas, estos algoritmos han logrado superar el desempeño de los profesionales médicos, como por ejemplo, en la detección de retinopatía diabética [10], la clasificación del cáncer de piel [2], entre otras.

En los últimos años, el diagnóstico automatizado a partir de radiografías de tórax fue tomando cada vez más relevancia, con algoritmos para la detección de neumonía [7], de nódulos pulmonares [4] y la clasificación de la tuberculosis pulmonar [21]. También se estudio el rendimiento de varias arquitecturas de redes neuronales convolucionales en diferentes anomalías [17].

Estudios sobre la utilización de imágenes de radiografía en la predicción de personas con COVID-19 son llevados a cabo por Wang y col. [26]. Presentan COVID-Net, un diseño de red neuronal convolucional profundo adaptado para la detección de casos de COVID-19 a partir de imágenes de rayos X de tórax. Estos diseños son de código abierto y están disponibles para el público en general. También presentaron un conjunto de datos de imágenes de referencia de acceso libre.

### 3 Extensión y alcance del proyecto

Con el fin de describir la extensión y el alcance del proyecto COVID-X a grandes rasgos, se resumen las responsabilidades que tiene el sistema desarrollado y sus principales implicaciones.

Para comenzar, se describe el componente o subsistema que debe generar o recolectar datos para el entrenamiento de la inteligencia artificial. Este tiene la responsabilidad de registrar datos cargados por usuarios del sistema para confeccionar el conjunto de entrenamiento.

En cuanto a los datos a registrar, entre ellos se incluye una radiografía de tórax. Las radiografías ingresarán al sistema a través del protocolo *DICOM* (*Digital Imaging and Communication On Medicine*) el cual especifica la definición de un formato de fichero, y un protocolo de comunicación de red. El formato de fichero incluye meta datos asociados al estudio además de la imagen propiamente dicha, con lo cual el sistema también debe ser capaz de procesar, almacenar y manipular este tipo de datos. A su vez, también se debe montar esta misma capacidad para registrar un conjunto de datos clínicos específicos que pueden ser adicionados por sobre el archivo DICOM de la radiografía.

En segundo lugar, como ya se ve la necesidad de manejar usuarios, se introduce un componente para la administración y control de usuarios del sistema e instituciones (a las que los usuarios pertenecen). Su responsabilidad también abarca la validación de datos de los mismos para poder garantizar la correcta construcción de esta colección de datos.

Adicionalmente, se decide incluir un subsistema que tenga la habilidad de ser consultado por usuarios especiales con el fin de generar conjuntos de entrenamiento con diversas características para la inteligencia artificial.

Finalmente, el sistema debe incluir dentro de sus funcionalidades los componentes necesarios para que un usuario del sistema pueda solicitar una inferencia (una predicción que realiza la inteligencia artificial). Por lo tanto, el usuario debe poder enviar los datos necesarios, estos son similares a los que se recolectan para armar el conjunto de datos de entrenamiento. Con lo cual, el sistema tiene que poder realizar predicciones a través de la inteligencia artificial previamente entrenada.

## 4 Producto desarrollado

El sistema COVID-X, brinda una herramienta basada en aprendizaje automático para la ayuda a médicos en el diagnóstico de la enfermedad COVID-19. A partir del análisis de datos clínicos y una radiografía de tórax, este genera una clasificación, reportando una inferencia con las probabilidades de que el paciente presente signos de neumonía por COVID-19, otros causales de neumonía viral y que no presente signo de ninguna afección relacionada al COVID-19. Para lograr esta clasificación, la inteligencia artificial de COVID-X no solo analiza los datos clínicos del paciente, sino que también detecta rasgos neumatológicos en las imágenes de rayos x.

Para llevar adelante el aprendizaje automático, el módulo de inteligencia artificial requiere ser nutrido de un vasto conjunto de casos ya analizados, es decir, información clínica y radiografía de tórax de pacientes que ya han sido diagnosticados de corona virus, así como también de pacientes con otras afecciones. La herramienta incluye la funcionalidad de carga de datos para el entrenamiento de la inteligencia artificial, que permite indicar si el paciente se realizó un hisopado RT-PCR (y sus resultados), si sufre de neumonía, EPOC, etc. Cuanto mayor sea la recolección de datos para entrenamiento, mejor serán los resultados de la clasificación de nuevos casos para los que se desee realizar una inferencia.

### 4.1 Requerimientos

Junto con la definición de los objetivos, alcance y extensión del proyecto, que se han descrito en secciones anteriores, se establecen los requerimientos o requisitos del sistema COVID-x.

#### 4.1.1 Requisitos funcionales

Se presenta en esta sección el listado de requerimientos funcionales del sistema COVID-X.

#### 4.1.1.1 Catálogo de requisitos

Número	Nombre	Tipo/Actor
1	Creación de un usuario	Usuario
2	Asignar roles a un usuario	Usuario
3	Ingreso de un usuario	Usuario
4	Consulta de información de un usuario	Usuario
5	Cambiar contraseña de un usuario	Usuario
6	Realizar carga de datos para entrenamiento de la IA	Usuario
7	Realizar carga de datos para una inferencia	Usuario
8	Consultar registros y métricas de uso del sistema	Sistema
9	Consultar historial de actividad de un usuario	Sistema
10	Verificar datos de contacto de un usuario nuevo	Sistema
11	Entrenar el módulo de IA	Sistema
12	Informar a los usuarios sobre novedades de la plataforma	Sistema
13	Validar datos de un usuario nuevo	Administrador
14	Desactivar o bloquear temporalmente a un usuario	Administrador

Tabla 1: Catálogo de requisitos funcionales.

1. El sistema debe permitir el registro y creación de usuarios: se debe poder dar de alta nuevos usuarios que pertenezcan a una dada organización (Hospitales, Clínicas, Sanatorios, Universidades, etc). Debe solicitar datos personales, de contacto y de la institución de pertenencia.
2. Se debe poder asignar roles a los usuarios. Algunos usuarios pueden subir datos para entrenamiento, realizar inferencias, ambas acciones, realizar búsquedas sobre los datos recolectados o administrar el sistema.
3. El sistema debe permitir el ingreso al mismo por parte de usuarios ya registrados, que han verificado sus datos de contacto y sus datos ya hayan sido correctamente validados por un moderador.
4. Se debe poder obtener los datos correspondientes a un usuario específico, si los roles del usuario que realiza la consigna incluyen el rol de administrador o se trata del mismo usuario.
5. El sistema debe permitir cambiar la contraseña de ingreso al sistema, a un usuario ya registrado, si este así lo deseara. Haciendo uso de los datos de contacto del usuario, se debe poder restablecer una contraseña de ingreso olvidada o simplemente cambiarla.

6. Se debe poder realizar la carga de datos para entrenamiento de la inteligencia artificial. Esto es, un usuario registrado, con datos de contacto verificados y validado por un moderador, que ha ingresado al sistema y tiene los roles necesarios, debe poder registrar tanto una radiografía de tórax en formato DICOM, como también una serie de datos clínicos opcionales para complementar la imagen. Como la inteligencia artificial implementa un algoritmo de aprendizaje automático supervisado, se debe incluir en información de carácter clínico si los datos registrados corresponden a un paciente COVID-19 positivo o negativo.
7. El sistema debe permitir a un usuario registrado, con datos de contacto verificados y validado por un moderador, que ha ingresado al sistema y tiene los roles necesarios, realizar una inferencia a partir de una radiografía de tórax complementada con los datos clínicos que se dispongan del paciente para el cual se quiere conocer la pérdida realizada por el sistema.
8. Se debe tener registros y métricas de uso del sistema, sobre la construcción del conjunto de entrenamiento, actividad de los usuarios y de las inferencias realizadas por el sistema para poder realizar reportes del sistema o aplicar técnicas analíticas sobre las métricas de uso del sistema.
9. Se debe poder informar a un usuario el historial de actividad (datos de entrenamiento cargados, inferencias realizadas). De igual forma, el sistema debe poder agrupar esta actividad por organización o institución para poder confeccionar tableros o reportes para poner a disposición de las mismas de ser necesario.
10. El sistema debe verificar los datos de contacto de un usuario nuevo. Al registrarse en el sistema, el usuario indica datos de contacto que deben ser verificados para poder utilizar este medio de contacto para información y notificaciones del sistema, así como también para validar su identidad si algún proceso lo requiere.
11. El sistema debe permitir entrenar y evaluar modelos de inteligencia artificial, utilizados para las inferencias, a partir de los datos recolectados para entrenamiento.
12. El sistema debe poder informar a los usuarios sobre novedades de la plataforma, la inteligencia artificial y la investigación constante del proyecto en forma de foro donde se detallen entradas con este tipo de información.

13. Se debe poder validar manualmente mediante un *backoffice* los datos de un usuario nuevo que ya ha validado sus datos de contacto. Con el uso del *backoffice*, una persona podrá validar que los datos ingresados sean correctos y que el nuevo usuario pertenezca efectivamente a la organización o institución que indico al registrarse.
14. El sistema debe permitir a un usuario administrador desactivar o bloquear temporalmente a un usuario en un contexto de moderación. El mismo *backoffice* anteriormente mencionado debe permitir realizar esta acción de forma preventiva.

#### 4.1.2 Requisitos no funcionales

Se presenta en esta sección el listado de requerimientos no funcionales del sistema COVID-X.

1. El sistema debe operar a nivel mundial. Dado que el objetivo del proyecto COVID-X es colaborar en el contexto de la pandemia del COVID-19, el sistema debe operar a nivel mundial y para ello debe soportar múltiples idiomas y particularidades.
2. El sistema debe garantizar la confidencialidad e integridad de la información de los usuarios.
3. El sistema debe garantizar la confidencialidad e integridad de la información de las organizaciones o instituciones participantes.
4. El sistema debe garantizar la confidencialidad e integridad de la información de datos de pacientes ingresada al sistema.
5. El sistema debe poseer trazabilidad de las acciones realizadas en el mismo. Se debe poder conocer quién, cuándo, dónde, y para qué organización se realiza cada operación que involucre manipulación de datos médicos.
6. El sistema debe estar disponible 24/7 debido a los distintos husos horarios. Además, en un contexto de emergencia sanitaria, la disponibilidad de esta herramienta se considera crucial.
7. El sistema debe ser fácil de usar. Los usuarios deben poder utilizar el sistema sin mayores inconvenientes debe ser intuitivo, debe usar terminología precisa para evitar ambigüedades dado que el público de usuarios del sistema estará principalmente compuesto por profesionales de salud o académicos que deberán proporcionar información de

forma certera. También debe ser fácil de usar para quienes actúen como clientes del sistema en integraciones *ad hoc* para ciertas instituciones o sistemas.

8. El sistema debe operar con exactitud y precisión, realizando inferencias con porcentajes de error lo más bajos posibles para evitar malas clasificaciones. Siendo esto último, crucial para el correcto uso y funcionamiento del sistema. La precisión es una componente fundamental y principal para el éxito del sistema, con la meta de minimizar la tasa de falsos negativos.
9. El sistema debe ser fácil de administrar, dado que el registro de nuevos usuarios requiere moderación manual, los administradores deben poder realizar estas moderaciones con facilidad para no generar demoras en proveer acceso al sistema a nuevos usuarios e instituciones.

## 4.2 Arquitectura

En el diseño de la arquitectura del sistema informático se utiliza la técnica de *Attribute Driven Design (ADD)*. Esta toma como base los requerimientos del sistema (tanto requerimientos funcionales como no funcionales), las restricciones del sistema y el conocimiento previo del problema. A partir de esta entrada, se derivan y definen los atributos de calidad de la arquitectura y los escenarios o drivers de la arquitectura.

### 4.2.1 Atributos de calidad

Se selecciona los atributos de calidad que se consideran más relevantes a la hora de diseñar y desarrollar la arquitectura del software correspondiente. Vale aclarar que aquellos atributos no mencionados se los considera simplemente de menor importancia, pero no implica que no apliquen y no se tengan en cuenta también en el desarrollo. A continuación, se detallan los atributos seleccionados en orden de prioridad:

1. Precisión
2. Seguridad
3. Escalabilidad
4. Disponibilidad
5. Usabilidad
6. Capacidad de Administración

#### 4.2.1.1 Precisión

El sistema debe poseer alto grado de exactitud y una precisión acorde al margen de error mínimo tolerado para la herramienta de asistencia en el diagnóstico médico. La precisión del sistema queda determinada por el error asociado a la clasificación inferida por la inteligencia artificial. En este contexto, para poder medir la precisión del sistema se analiza los resultados de las inferencias con la métrica de sensibilidad. Se busca maximizar esta métrica con el fin de minimizar el riesgo de los falsos negativos, que son un problema importante en el ámbito médico.

Dado el potencial infectivo del virus SARS-CoV-2, los falsos negativos representan una amenaza de rebrote que atenta contra la capacidad del sistema sanitario de un país. Visto de forma más granular, provocaría que quienes lo poseen y sean indebidamente diagnosticados no sean tratados y no queden aislados, convirtiéndose así en agentes propagadores del virus.

#### 4.2.1.2 Seguridad

Para poder garantizar la integridad y confidencialidad de los datos sensibles de los usuarios de la plataforma, ya sean datos personales, institucionales, clínicos o cualquier otro dato sensible, la seguridad es un pilar fundamental en el diseño de la arquitectura del sistema. El sistema no debe presentar vulnerabilidades y contemplar riesgos de seguridad. Debe proponer mecanismos de contingencias ante todos los ataques comúnmente conocidos derivados de explotar las vulnerabilidades especificadas en proyectos como *OWASP (Open Web Application Security Project)*.

En lo que respecta a la capacidad de auditar el sistema, una consideración no menor es poder mantener una traza de todas las acciones que se llevan a cabo de forma que sea posible realizar un análisis posterior que permita averiguar lo ocurrido bajo cualquier escenario. En la trazabilidad se busca poder determinar quién, cuándo, y dónde se realiza cada acción, en especial en aquellas que involucran lectura, o escritura de datos clínicos.

#### 4.2.1.3 Escalabilidad

El sistema de COVID-X opera de forma internacional y se integra con instituciones médicas u organizaciones de la salud. En el contexto de la emergencia sanitaria, al igual que el aumento de los casos potenciales de COVID-19, la demanda de asistencia al diagnóstico puede crecer exponencialmente. Se contempla la demanda altamente variable que implica el uso de la plataforma por diversas instituciones en distintos países. En estos ca-

El sistema debe poder escalar para manejar una cantidad variable de transacciones.

#### **4.2.1.4 Disponibilidad**

Dado que se operará en todo el mundo y por lo tanto se trabaja con distintos husos horarios, el sistema debe estar siempre disponible (24 horas al día 7 días a la semana). Por este motivo se lo considera un atributo de alta relevancia, dado que si el sistema no está disponible, los usuarios no podrán efectuar predicciones para apoyar el diagnóstico preventivo de los potenciales pacientes infectados por el virus.

#### **4.2.1.5 Usabilidad**

Tratándose de un sistema con interacción directa con los usuarios, la facilidad de uso resulta un atributo de calidad imprescindible. La interfaz gráfica con la que interactúa el usuario debe ser intuitiva, fácil de aprender, y didáctica.

Por otra parte, la usabilidad es muy importante para las integraciones *ad hoc* realizadas con ciertas instituciones u organizaciones. En el caso que una organización requiere una integración a medida para incorporar en sus sistemas o para operar con COVID-X directo desde sus equipos sin utilizar la interfaz web provista, esta integración debe ser intuitiva con una adecuada documentación y canales de soporte.

#### **4.2.1.6 Capacidad de Administración**

Por último, la capacidad del sistema de ser administrado es un atributo destacable a tener en cuenta. Como se menciona en los requerimientos, el sistema debe contar con moderación manual para validar los datos de nuevos usuarios registrados que han verificado sus datos de contacto, así como también moderar usuarios ya validados y nuevas organizaciones. Con esto en mente, la facilidad de realizar estos procesos manuales impacta sobre la rapidez de estos procesos y en particular, para los nuevos usuarios se busca minimizar el tiempo de demora para la validación y activación de su cuenta.

### **4.2.2 Modelo de vistas**

Para describir la arquitectura del sistema, se hace uso de múltiples vistas concurrentes. Estas fueron introducidas por Kruchten [20], describiendo el modelo "4+1". Este uso de múltiples vistas permite describir la arquitectura

desde el punto de vista de los diferentes interesados: usuario final, desarrolladores, ingenieros de sistemas, gerentes de proyecto, etc., y manejar por separado los requisitos funcionales y no funcionales.

Las cuatro vistas del modelo son: vista lógica, vista de desarrollo, vista de proceso y vista física. Además, una selección de casos de uso o escenarios se utiliza para ilustrar la arquitectura sirviendo como una vista más. Las vistas se diseñan utilizando un proceso de desarrollo iterativo, basado en escenarios y centrado en la arquitectura. A continuación, se describe cada una de las cinco vistas.

#### 4.2.2.1 Vista lógica

Para exponer esta vista, enfocada en describir la estructura y funcionalidad del sistema, se presenta un diagrama en bloques de alto nivel del sistema COVID-X:

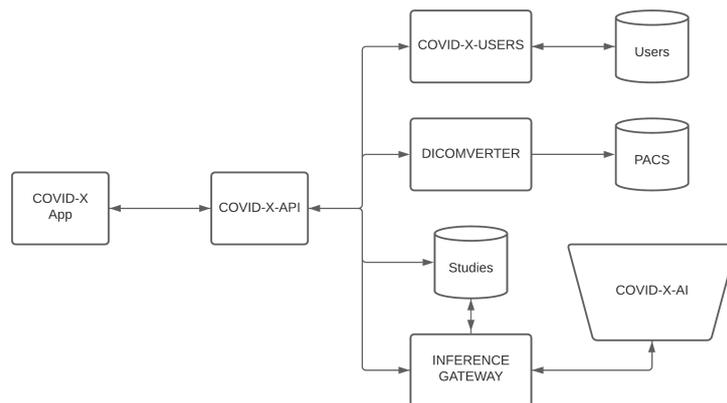


Figura 1: Diagrama en bloques del sistema COVID-X.

Como se puede observar en el diagrama de la figura 1, el punto de entrada al sistema es la aplicación web, donde los usuarios finales del sitio (médicos, investigadores, etc) pueden realizar tanto una carga de datos para entrenamiento como consultar una inferencia sobre un paciente.

El *backend* del sistema, se encuentra orquestado por un componente principal llamado COVID-X-API que es el encargado de coordinar las operaciones antes mencionadas. COVID-X-API es una interfaz de programación de aplicaciones (API por sus siglas en inglés) que se comunica con otros bloques o módulos.

El modulo COVID-X-USERS engloba todo el manejo de usuarios del sistema, almacenando y resguardando la información de los mismos y con un

estricto control de acceso para los recursos del sistema.

Por otra parte, DICOMVERTER se encarga del procesamiento y tratamiento de las imágenes DICOM proporcionadas por el usuario, almacenando las imágenes DICOM anonimizadas en el servidor *PACS (Picture Archiving and Communication System)*.

Los estudios son una abstracción de los casos analizados o de entrenamiento, representan la unión de un archivo DICOM y la información clínica ingresada en un formulario desde la aplicación web al momento de generar una carga de entrenamiento o realizar una inferencia.

El INFERENCE GATEWAY es el nexo entre los estudios y modelo utilizado de la inteligencia artificial COVID-X-AI, es quién realiza las operaciones necesarias para llevar a cabo una inferencia.

El modelo de inteligencia artificial COVID-X-AI, es principalmente una composición entre un modelo redes neuronales profundas y uno basado en arboles de decisión. Se utiliza un modelo jerárquico en el que la información es pre clasificada entre posibles casos con rasgos de neumonía y aquellos que no presentan signos. Luego se aplica una clasificación más granular donde se diferencia entre signos de neumonía por COVID-19 versus signos provocados por otros causantes. Por ultimo, junto a este resultado se lo opera en conjunto al extraído con los arboles de decisión basados en información clínica, dando así el resultado final.

#### **4.2.2.2 Vista de proceso**

En esta vista, se representan los flujos de trabajo del sistema paso a paso. Se exponen los dos procesos principales del sistema COVID-X mediante diagramas de secuencia. Se opta por el uso de diagramas de secuencia por sobre los de actividad debido a que los primeros se enfocan en el comportamiento del sistema en tiempo de ejecución, brindando mayor nivel de detalle. Mientras que los segundos, tienen un alto nivel de abstracción.

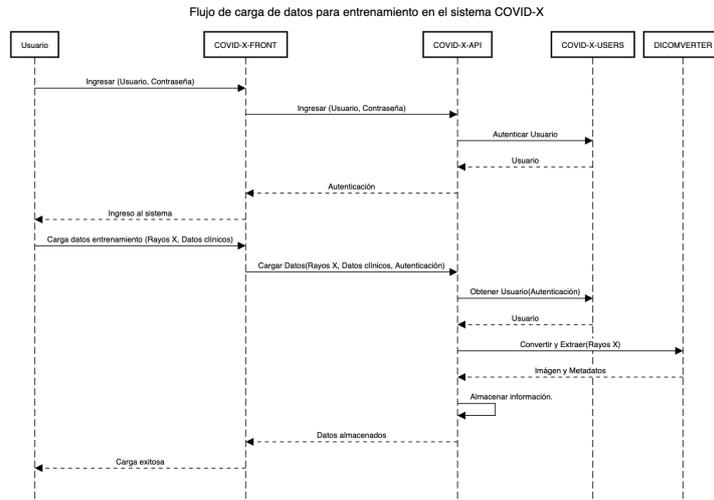


Figura 2: Diagrama de secuencia de la carga de datos para entrenamiento en el sistema COVID-X.

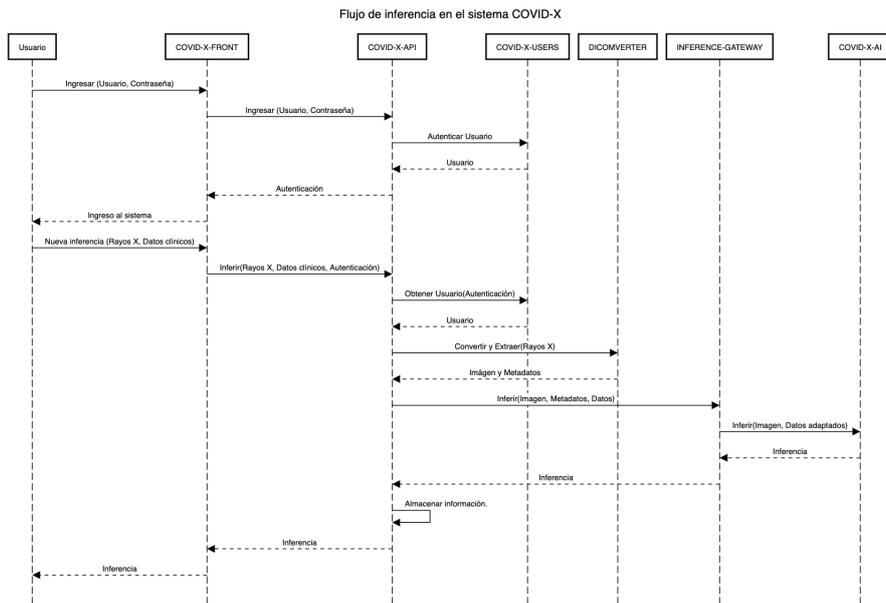


Figura 3: Diagrama de secuencia del flujo de inferencia en el sistema COVID-X.

#### 4.2.2.3 Vista de desarrollo

La vista de desarrollo representa la organización real de los módulos de software en el entorno de desarrollo. Para esquematizar los módulos de la

arquitectura desde la perspectiva de desarrollo, se exponen diagramas de lenguaje unificado de modelado UML (Unified Modeling Language por sus siglas en ingles). Estos permiten visualizar, especificar, construir y documentar un sistema, además de que es conocido como un estándar en la documentación y especificación de sistemas.

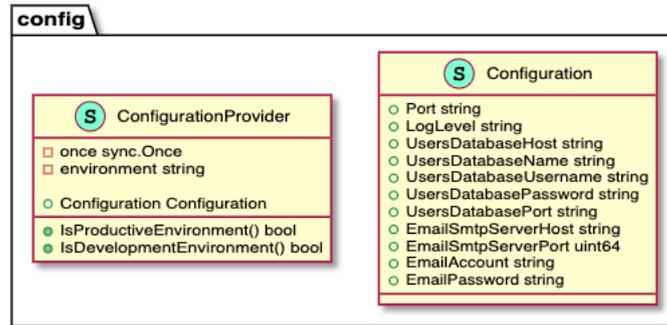


Figura 4: Diagrama UML con la configuración requerida por COVID-X-USERS.

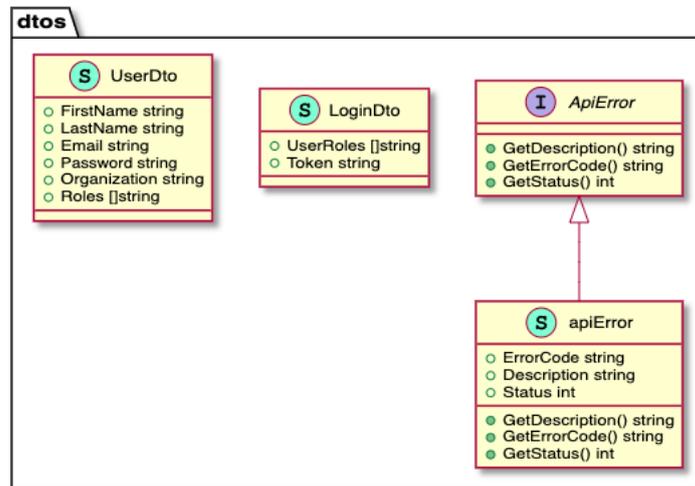


Figura 5: Diagrama UML con los objetos de transferencia de datos de COVID-X-USERS.

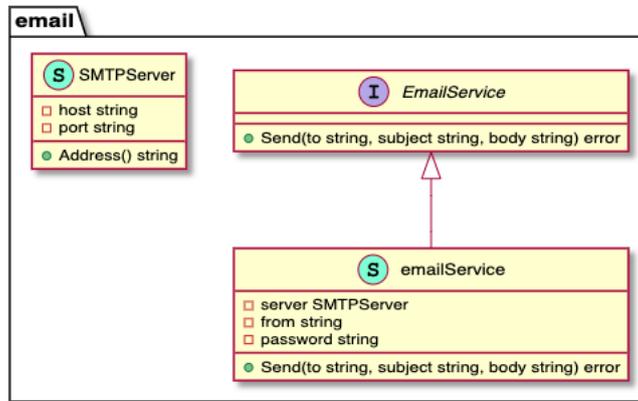


Figura 6: Diagrama UML con los servicios de correo electrónico de COVID-X-USERS.

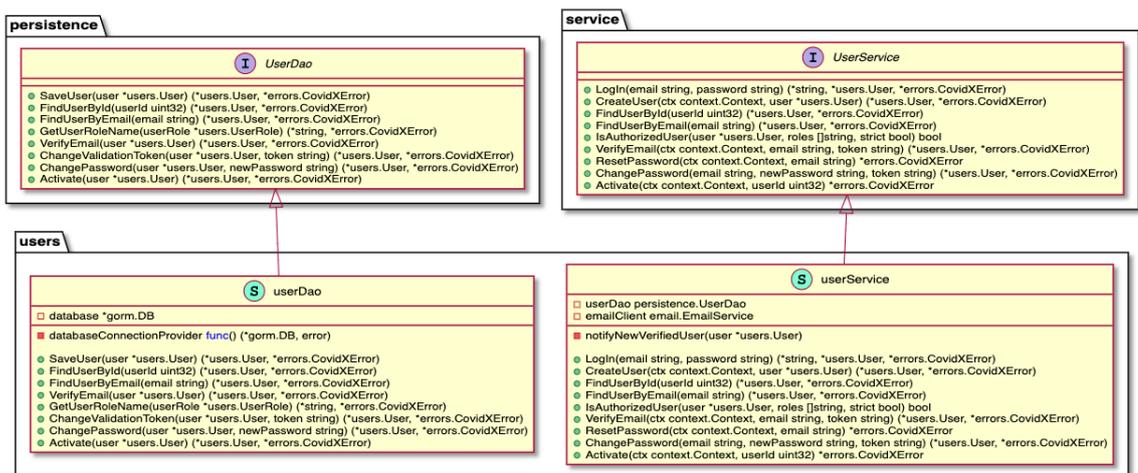


Figura 7: Diagrama UML de servicios y persistencia de COVID-X-USERS y sus implementaciones.

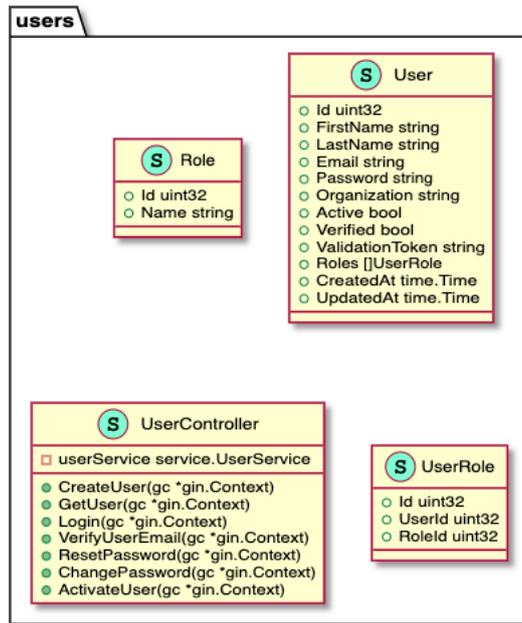


Figura 8: Diagrama UML de entidades y controlador de COVID-X-USERS.

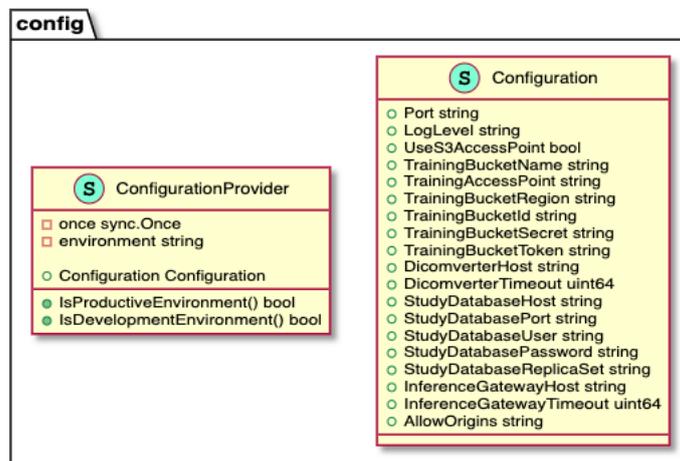


Figura 9: Diagrama UML con la configuración requerida por COVID-X-API.

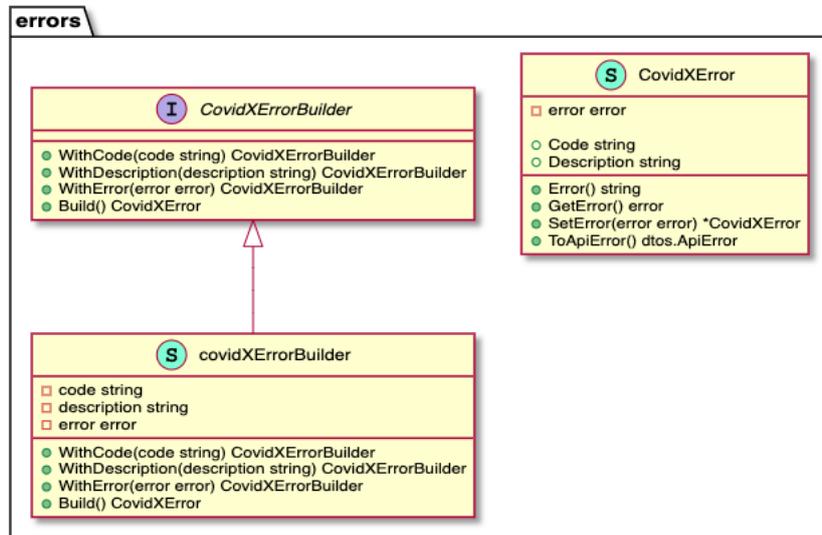


Figura 10: Diagrama UML con los objetos de errores de COVID-X-API.

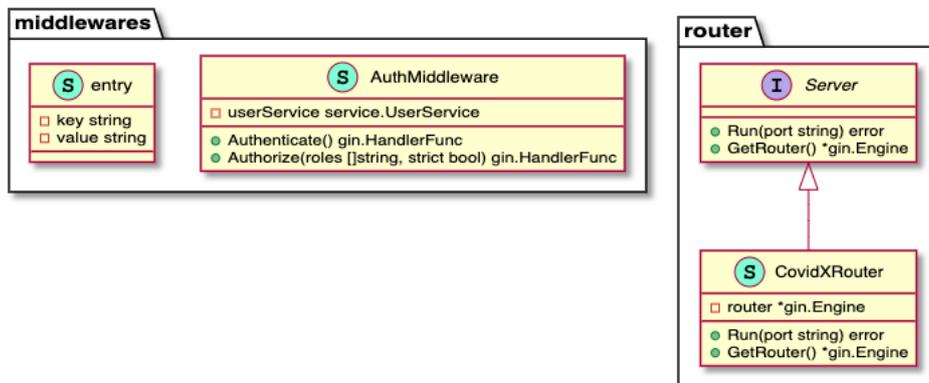


Figura 11: Diagrama UML con objetos de soporte HTTP de COVID-X-API.

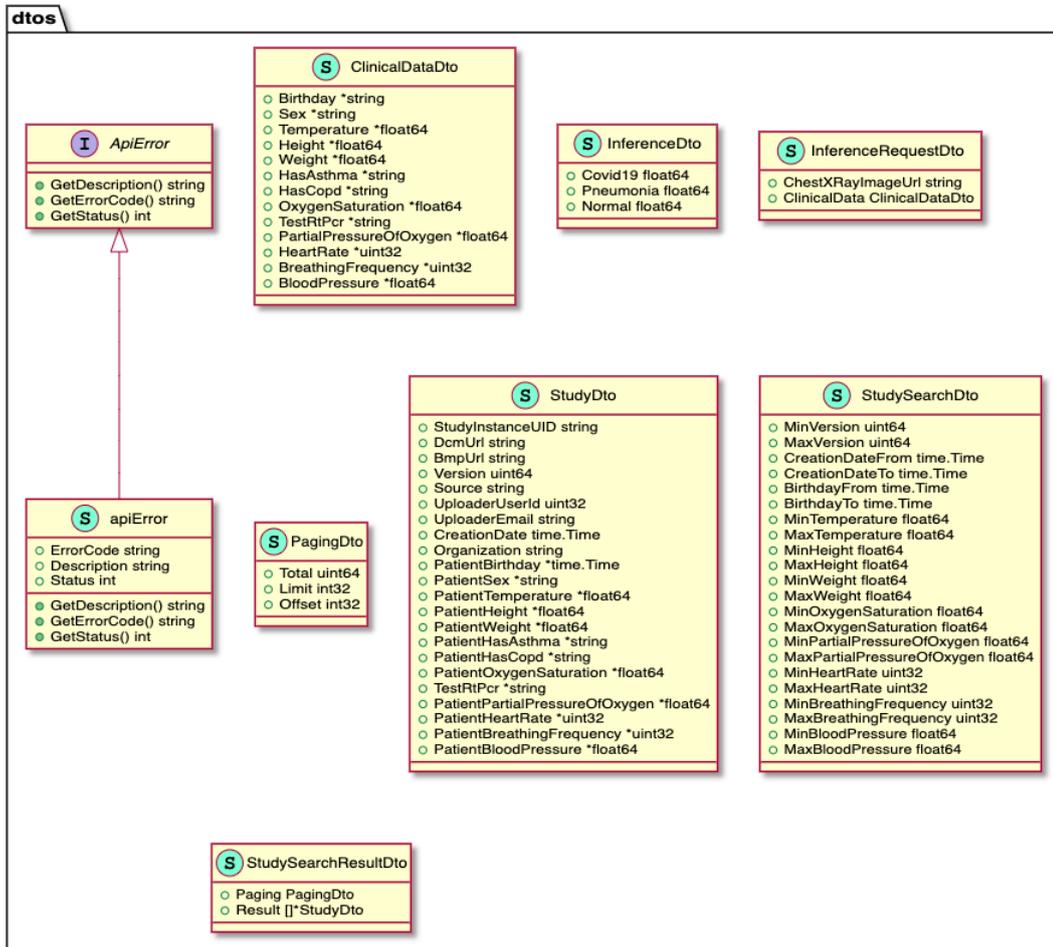


Figura 12: Diagrama UML con los objetos de transferencia de datos de COVID-X-API.

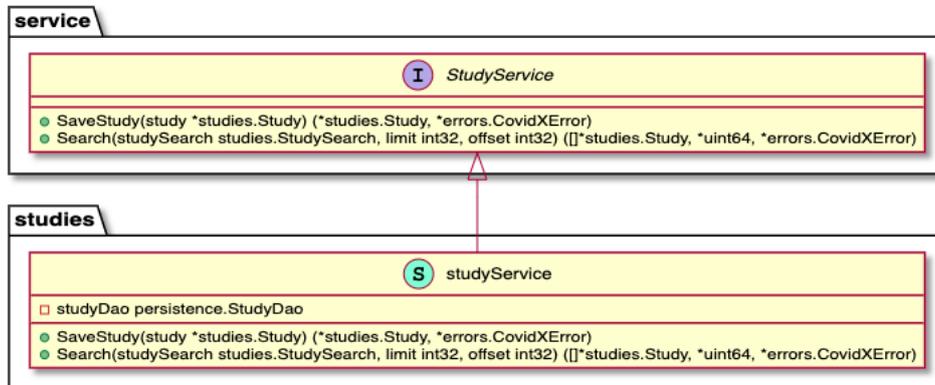


Figura 13: Diagrama UML de los servicios de estudios de COVID-X-API.

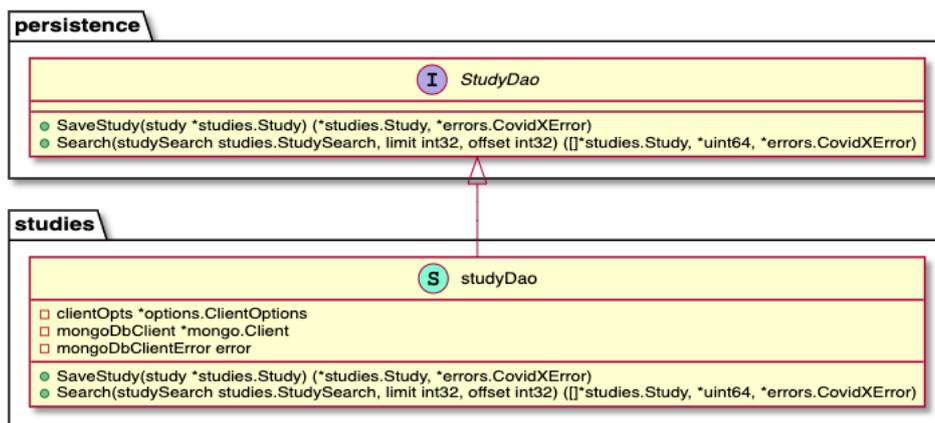


Figura 14: Diagrama UML de la persistencia de estudios de COVID-X-API.

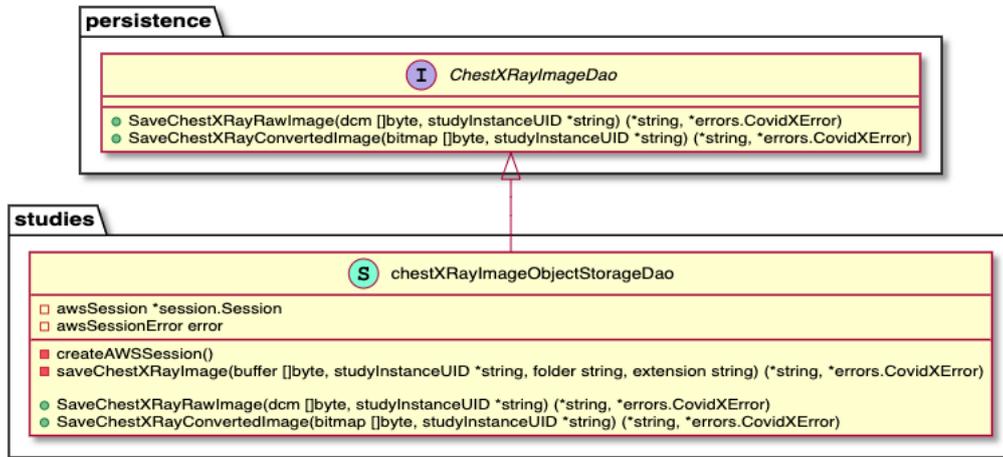


Figura 15: Diagrama UML de la persistencia de imágenes radiográficas de tórax de COVID-X-API.

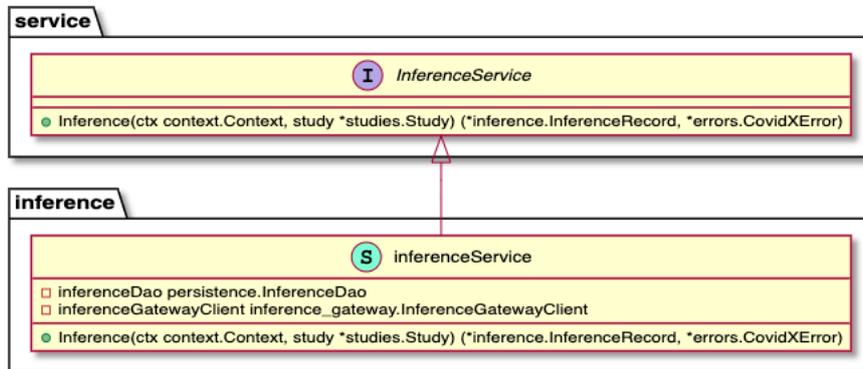


Figura 16: Diagrama UML de los servicios de inferencias de COVID-X-API.

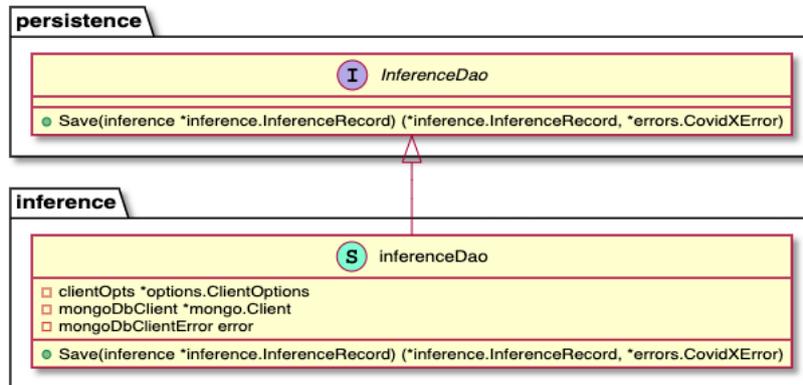


Figura 17: Diagrama UML de la persistencia de inferencias de COVID-X-API.

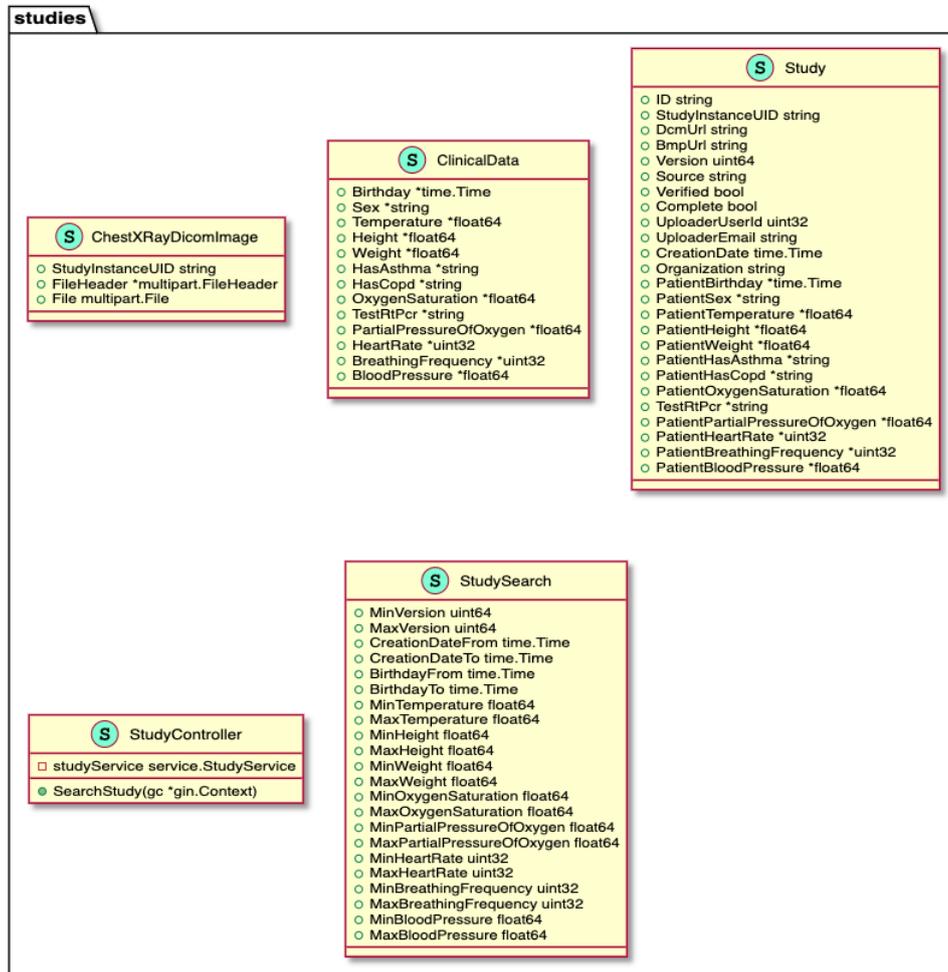


Figura 18: Diagrama UML con las entidades y el controlador de estudios de COVID-X-API.

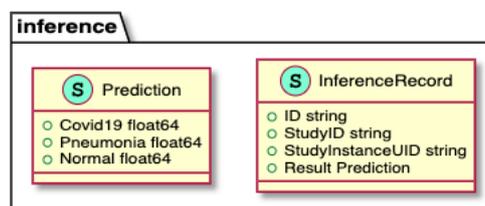


Figura 19: Diagrama UML con las entidades de inferencia de COVID-X-API.

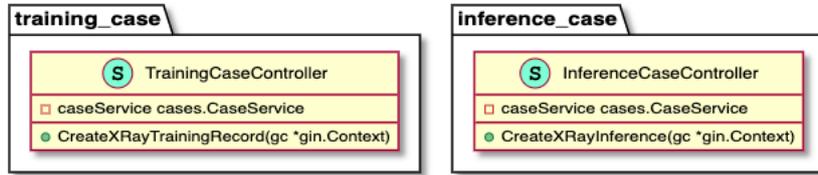


Figura 20: Diagrama UML con los servicios de los flujos de inferencia y entrenamiento de COVID-X-API.

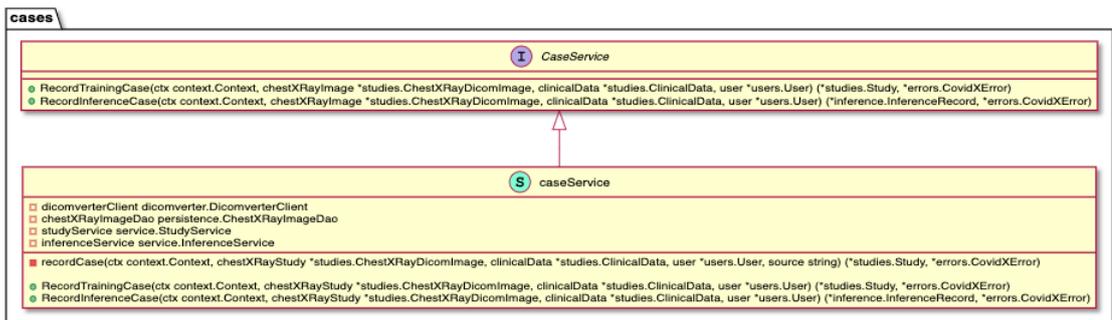


Figura 21: Diagrama UML con los servicios de los flujos de COVID-X-API.

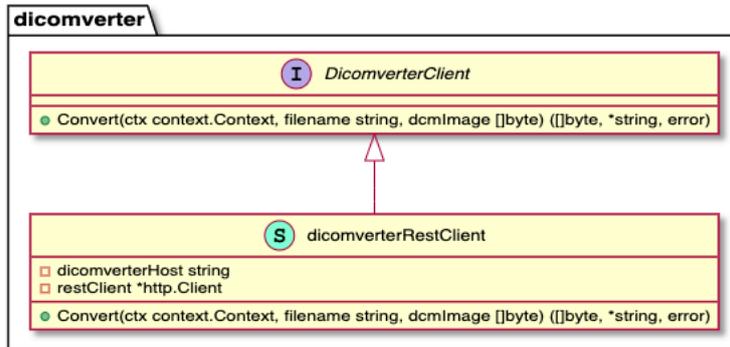


Figura 22: Diagrama UML con el cliente de DICOMVERTER utilizado en COVID-X-API.

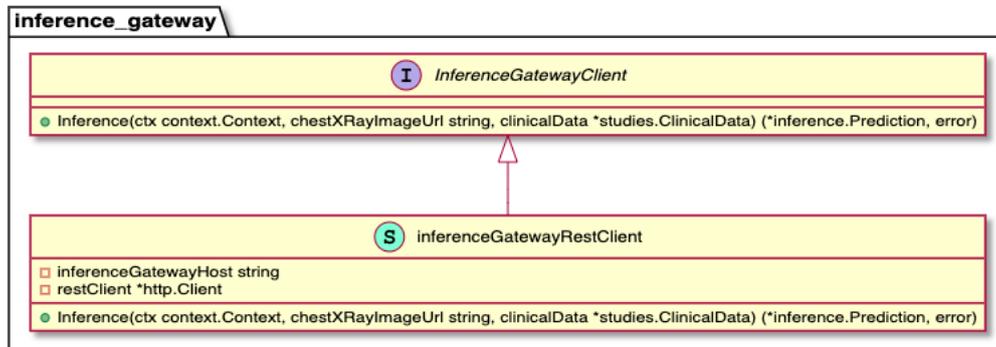


Figura 23: Diagrama UML con el cliente de INFERENCE GATEWAY utilizado en COVID-X-API.

#### 4.2.2.4 Vista física

Otra de las perspectivas desde las cuales se describe la arquitectura de COVID-X es la vista física. Esta representa como están distribuidos los componentes entre los distintos módulos que conforman el sistema informático.

En la Figura 24, se presenta un diagrama de despliegue que pretende describir esta vista. El diagrama se esquematiza en el ecosistema de *AWS (Amazon Web Services)*, dado que es el proveedor de nube pública provisto por el ITBA, utilizado para montar el sistema COVID-X.

El sistema está diseñado para que el punto de entrada al mismo por parte de los usuarios finales sea a través de una aplicación web desplegada en el servicio *AWS Amplify*, siendo posible acceder a esta a través el dominio *cov-x.com* utilizando el servicio de DNS *AWS Route53*. Esta aplicación web se comunica con COVID-X-API que se encuentra al igual que los demás servicios REST en contenedores de *Docker* orquestados por *AWS ECS (Elastic Container Service)* con *AWS Fargate*. En cambio, los administradores pueden acceder a los recursos del sistema vía el *bastion host*, de esta manera, se logra ofrecer seguridad a las redes internas obteniéndose una arquitectura mas robusta y segura.

En frente de los contenedores de COVID-X-API, COVID-X-USERS, DICOMVERTER, INFERENCE-GATEWAY, y la instancia de BASTION-HOST se presentan balanceadores de carga a nivel de aplicación, actuando como único punto de entrada a ellos. Los balanceadores de carga distribuyen el tráfico entrante de aplicaciones entre varios destinos, aumentando la disponibilidad de la aplicación. Para escalar horizontalmente los servicios, se combinan los balanceadores de carga con los *Auto Scaling Group de AWS*, permitiendo escalar y des-escalar el numero de instancias de los servicios *stateless*

en respuesta a la demanda y con las restricciones de su configuración. Para administrar el escalamiento se utiliza el servicio *AWS ECS*.

Como se aprecia en el diagrama, se trata de una arquitectura basada en microservicios. Este es un enfoque para desarrollar un sistema como un conjunto de pequeños servicios, cada uno ejecutándose de forma autónoma, correspondiendo a una responsabilidad o área de negocio particular, y comunicándose entre sí. Los módulos de servicios de COVID-X se comunican entre si utilizando canales de red privados y mediante el protocolo HTTPS, para brindar mayor seguridad.

Para el almacenamiento de las imágenes convertidas a BMP de radiografías de tórax, se utiliza el servicio de *Amazon Simple Storage Service o S3*. Este *object storage* o *bucket* se encuentra configurado para que solo pueda ser accedido desde COVID-X-API y COVID-X-TRAINER utilizando los *IAM Roles* y políticas de *AWS*. También se guardan las imágenes originales en formato DICOM, como resguardo o *backup* de lo almacenado en el servidor COVID-X-PACS. En cuanto a los modelos entrenados por COVID-X-TRAINER, se almacenan en otro *S3* que solo puede ser accedido por COVID-X-TRAINER e INFERENCE-GATEWAY. El servicio *AWS S3* es un servicio global dentro de *AWS*, y el mismo administra su disponibilidad.

Luego para el manejo de usuarios, el modulo de COVID-X-USERS hace uso de una base de datos *PostgreSQL* administrada a través del servicio de *Amazon Relational Database Service (RDS)*. El motivo de su elección fue para desligarse de las implicaciones que lleva el mantenimiento del motor de base de datos. En lo que respecta a la persistencia de estudios, se utiliza un servicio provisto por *Mongo Atlas* por los mismos motivos por los cuales se utiliza *AWS RDS*, esto es, evitar cualquier tipo de mantenimiento y facilitar su administración.

Además, los módulos de COVID-X pueden ser monitoreados mediante el uso de *Amazon CloudWatch Logs* que permite llevar un registro de logs de las instancias y demás servicios de *Amazon* que se utilizan.

En las sección 4.4, se explican mas detalladamente las responsabilidades y funciones de cada uno de estos servicios, en 4.5 la aplicación web, y finalmente se describen mas detalles de COVID-X-TRAINER en la sección 4.3.4.1.

La arquitectura se encuentra montada sobre la nube publica de *AWS*, para crear, administrar y configurar toda la infraestructura mencionada en la vista física, se utiliza la herramienta *Terraform*. *Terraform* es un software de infraestructura como código que permite definir y configurar la infraestructura en un lenguaje de alto nivel, generando un plan de ejecución para desplegar la infraestructura un proveedor de nube publica, en este caso *AWS*. En la sección 4.6 se presentan más detalles de la infraestructura del sistema.

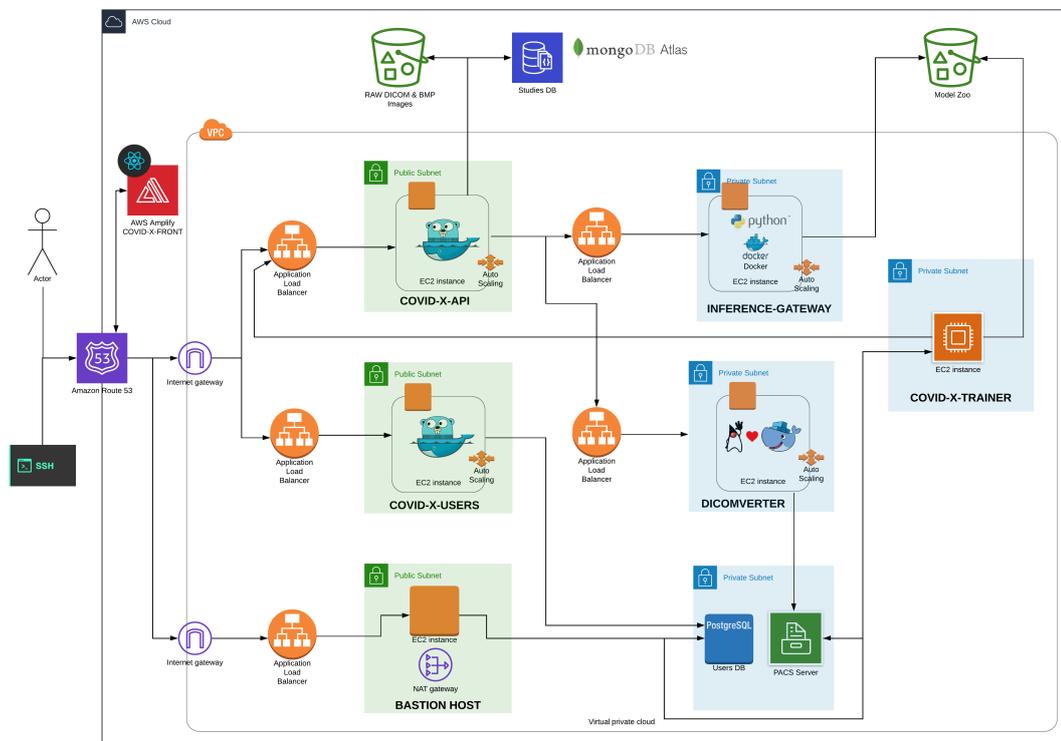


Figura 24: Diagrama de despliegue del sistema COVID-X.

#### 4.2.2.5 Escenarios

La metodología o técnica *ADD* propone analizar la arquitectura candidata frente a los escenarios propuestos. Si la misma no cumple un escenario satisfactoriamente, se debe introducir un cambio en la arquitectura y revalidar la nueva arquitectura iterando sobre todos los escenarios. Para llegar a la arquitectura se realizaron varias iteraciones. A continuación se presentan los escenarios que componen esta última vista o perspectiva de la arquitectura de COVID-X.

1. Alta demanda del sistema
2. Caída de alguna base de datos
3. Ingreso de datos fraudulentos
4. Exposición de datos sensibles
5. Caída del servidor PACS

6. Pérdida de información

7. Caída de una zona o región del proveedor de nube pública

#### **4.2.2.6 Escenario 1: Alta demanda del sistema**

Este escenario describe que dada la integración de varias instituciones médicas de gran magnitud, que concurrentemente, en un período particular durante el colapso del sistema sanitario, utilicen el sistema. Este escenario se contempla mediante la escalabilidad horizontal de todos los componentes de la arquitectura basada en microservicios, a su vez, se suma las de réplicas para lectura de las bases de datos. Por otra parte los servicios externos también tienen un contrato de servicio que garantiza su correcto funcionamiento bajo un nivel de servicio establecido y escalabilidad consistente con nuestro sistema en torno a este escenario.

#### **4.2.2.7 Escenario 2: Caída de alguna base de datos**

Un escenario común y potencial donde se produce una caída o falla de alguna de las bases de datos. Las contingencias para este escenario son parte de la arquitectura misma. En el caso de la base de datos de usuarios, se mitiga con la replicación *master-slave* en las bases de datos, si un nodo *master* llegara a fallar, el sistema le otorgaría su rol a el nodo *slave* hasta su recuperación. Esta base de datos *OLTP* (*OnLine Transaction Processing*) evita la pérdida de datos implementando *Write Ahead Logging* (*WAL*) y tampoco se pierde disponibilidad de la misma.

En cuanto a la base de datos de estudios e inferencias, como se mencionó es una base de datos no relacional orientada a documentos, en particular *MongoDB*, configurada con un *replica set* y un sistema de replicación y entre sus nodos. Es decir, si se produce una baja del servicio de un nodo *master*, otro pasa a ser el nuevo nodo *master*. Las réplicas están ubicadas en distintas *Availability Zones* de *AWS*.

En lo que respecta a los *object storage*, se dispone de un contrato de servicio como se mencionó anteriormente.

#### **4.2.2.8 Escenario 3: Ingreso de datos fraudulentos**

Este escenario abarca la incorporación de datos que formarán parte del conjunto de entrenamiento que no sean verídicos, pudiendo afectar el entrenamiento de nuevos modelos de la IA. Para evitar esto, se incorpora en la solución un riguroso proceso de verificación y validación de información de usuarios e instituciones que forman parte de COVID-X. Por otra parte,

además de la validación previa para nuevos usuarios, los moderadores del sistema pueden desactivar o bloquear temporalmente cuentas de usuarios cuyas acciones comprometan el estado del sistema.

En adición a esto, todos los estudios tienen un usuario dueño o responsable de introducir esos datos. Por lo tanto, se pueden fácilmente excluir del conjunto de datos de entrenamiento todos los provistos por cualquier usuario que este involucrado en algún situación similar a las descritas en este escenario.

#### 4.2.2.9 Escenario 4: Exposición de datos sensibles

En la gran mayoría de los sistemas informáticos, garantizar la confidencialidad e integridad de la información es sumamente importante. En el caso de COVID-X, no es la excepción. Más aún, en este caso, donde se trata con datos clínicos de pacientes se debe cumplir con el deber de secreto.

Para cumplir con este escenario, donde se prevé una amplia gama de posibles situaciones exposición o filtrado de datos sensibles. La arquitectura busca prevenir de manera activa esta situaciones desde un enfoque global que alcanza a todos los módulos y subsistemas. Desde las bases de datos hasta las comunicaciones de los servicios todo prevalece seguro.

En materia de confidencialidad, se encripta la información de los usuarios así también como los datos clínicos de los pacientes siguiendo los estándares y guías de cifrado propuestos por *NIST*. De igual forma, la integridad de los datos sensibles se logra complementando la técnicas de cifrado con métodos de firma digital. Comenzando por la persistencia de los datos, la base de usuarios *PostgreSQL* se encripta con haciendo uso de los mecanismos de cifrado provistos por el servicio *AWS RDS*. Luego la base de estudios e inferencias, el servicio *Mongo Atlas*, al igual que *RDS*, permite configurar "Encryption at Rest", cifrando todos los *cluster* y todos los *snapshots* realizados en los procesos de resguardo de información.

En el caso de las imágenes, *Amazon S3* es seguro de forma predeterminada. Se crean usuarios con distintos roles para que cada servicio pueda interactuar con los privilegios mínimos requeridos. Se implementan con ellos mecanismos de control de acceso, como por ejemplo las políticas de *bucket* y las listas de control de acceso (*ACL* por sus siglas en ingles) para conceder permisos de manera selectiva a los usuarios y grupos de usuarios. Se utiliza autenticación de usuario para controlar el acceso a los datos, y así además lograr una buena trazabilidad.

Se configura el *bucket* de *Amazon S3* para que cree registros de acceso por todas las solicitudes creadas y destinadas a este. Además, se puede obtener información sobre los registros con *AWS CloudTrail Data Events*.

Para lo que refiere a "Encryption in transit", como se mencionó anteriormente, todas las comunicaciones internas entre los microservicios están aseguradas bajo el protocolo HTTPS. La comunicación entre los usuarios y el sistema, se encuentra asegurada de igual forma.

Finalmente, dentro del proveedor de nube pública, se tienen las consideraciones de la configuración de las redes, reglas de enrutamiento, cortafuegos, roles y políticas de uso y acceso.

#### **4.2.2.10 Escenario 5: Caída del servidor PACS**

En este escenario se describe una situación de falla o caída del servidor PACS de COVID-X.

El diseño de la arquitectura tiene en cuenta que soporte contingencias en los puntos no tan críticos para los flujos principales. Consecuentemente, el sistema tolera fallos o caídas del PACS por construcción.

Además, el servidor PACS internamente se implementa con una base de datos relacional *PostgreSQL*, por lo que aplica también lo mencionado para la base de datos de usuarios en el Escenario 2.

#### **4.2.2.11 Escenario 6: Pérdida de información**

La información es de suma importancia en el sistema COVID-X. En particular, para el entrenamiento de los modelos de la IA. Por ello, la arquitectura contempla minimizar los riesgos de pérdida de información. En lo que respecta a los datos de usuario, el servicio *AWS RDS* se configura para resguardar la información realizando y almacenando *snapshots* periódicos de las bases de datos. Al igual que el servicio *AWS RDS*, Mongo Atlas en el caso de estudios e inferencias, facilita la administración del sistema para resguardar la información permitiendo *backups* automatizados y hasta incluso realizarlos utilizando *storage services* de *AWS* para estos.

En lo que respecta al servidor PACS, las imágenes originales DICOM procesadas por el sistema, se resguardan en un *AWS S3*, y todas usan como nombre un id único relacionado con la radiografía y con el registro en la base de datos de estudios de COVID-X.

Adicionalmente, se utiliza el control de versiones de *AWS S3* que permite conservar, recuperar y restaurar todas las versiones de todos los objetos almacenados en un *bucket*. En caso de que se realice una operación de borrado sobre una imagen DICOM en *AWS S3*, se producirá una baja lógica, conservándose todas las versiones de dicha imagen en el *bucket*, y estas podrán recuperarse o restablecerse.

#### **4.2.2.12 Escenario 7: Caída de una zona o región del proveedor de nube pública**

Al integrarse con un solo proveedor de nube pública, no se administra la infraestructura y existe un modelo de responsabilidades para cada una de las partes en la contratación de este tipo de servicio. También este esquema, asume la existencia de una probabilidad no nula de la baja en algún servicio contratado.

Como *AWS* en su infraestructura global opera con zonas de disponibilidad agrupadas por regiones y compuestas por centros de cómputos de alta disponibilidad y redundancia de recursos, es infrecuente que se produzcan errores que afecten a la disponibilidad de las instancias que están en una misma región.

Actualmente, el sistema está alojado en una única región de *AWS* para evitar cargos extras en la facturación. Pero para mitigar el riesgo de la caída de una región de *AWS*, se puede fácilmente desplegar un esquema de múltiples regiones que garantice la disponibilidad, aumente la tolerancia a fallos y la estabilidad del sistema. Esto se puede lograr, gracias a, como se ha mencionado, la utilización de *Terraform* para administrar la infraestructura. Permitiendo automatizar y replicar rápidamente toda la infraestructura del sistema en cualquier otra región de *AWS*.

### **4.3 Inteligencia artificial**

El modelo de inteligencia artificial utilizado es una arquitectura compuesta, la cual tiene como componente principal una red neuronal artificial convolucional. Esta red se utiliza para la clasificación de imágenes de radiografía de tórax. De forma complementaria se incorpora un modelo de aprendizaje automático que utiliza el método *Random Forest* con datos clínicos de un paciente. Para obtener el resultado final se calcula el promedio pesado a partir de los resultados parciales de ambos modelos, donde los pesos son una variable a determinar por aprendizaje del sistema.

#### **4.3.1 Marco conceptual**

A continuación se describen los conceptos relacionados con los modelos de IA utilizados en la implementación de COVID-X.

##### **4.3.1.1 Redes neuronales convolucionales**

Las redes neuronales convolucionales son una variación de un perceptrón multicapa, y son muy efectivas para tareas de visión artificial, como en la

clasificación y segmentación de imágenes. La arquitectura de estas redes se compone por múltiples capas de filtros convolucionales. Para clasificación, al principio se dispone la fase de extracción de características, formada de neuronas convolucionales y de reducción de muestreo. En el final de la red se encuentran neuronas con función de activación logística para realizar la clasificación final sobre las características extraídas. Esta parte se compone tanto de capas de neuronas convolucionales como de neuronas de reducción de muestreo. Al progresar los datos a lo largo de estas capas, se disminuye su dimensión, siendo las neuronas en capas lejanas mucho menos sensibles a cambios en los datos de entrada.

En aprendizaje de redes profundas, una red neuronal artificial convolucional es una clase de red profunda que es frecuentemente utilizada en tareas de visión computarizada. Una característica distintiva las CNNs es su capacidad para explotar la correlación espacial o temporal en los datos. Su topología abarca múltiples etapas de aprendizaje compuestas por una combinación de: capas convolucionales, unidades de procesamiento no lineales, y capas de submuestreo [19]. CNN es una red jerárquica de múltiples capas de retroalimentación, donde cada capa, utilizando un banco de *kernels* convolucionales, realiza múltiples transformaciones [22].

La operación de convolución ayuda en la extracción de características útiles de puntos de datos correlacionados localmente. La salida de los *kernels* convolucionales se asigna luego a la unidad de procesamiento no lineal (función de activación), que no solo ayuda a aprender abstracciones sino que también integra la no linealidad en el espacio de características. Esta no linealidad genera diferentes patrones de activaciones para diferentes respuestas y, por lo tanto, facilita el aprendizaje de las diferencias semánticas en las imágenes. La salida de la función de activación no lineal suele ir seguida de un submuestreo, que ayuda a resumir los resultados y también hace que la entrada sea invariable a las distorsiones geométricas.

#### 4.3.1.2 Random Forest

El método de aprendizaje automático de *Random Forest* consiste en aplicar el método de *bagging* con una muestra menor o igual al conjunto de entrenamiento y para cada subconjunto de datos se construye un árbol de decisión correspondiente. Luego, el resultado final es el resultado "más votado" por todos los árboles anteriores.

El método de *bagging* (o *Bootstrap Aggregating*) consiste en optimizar predicciones utilizando remuestreo uniforme con reemplazo. Esta técnica consiste en crear diferentes modelos usando muestras aleatorias con reemplazo y luego combinar los resultados. Para ello se divide el conjunto de

entrenamiento en subconjuntos de datos. Luego se construye un modelo predictivo con cada conjunto, obteniendo modelos diferentes. A partir de ellos, se crea un único modelo, que es una síntesis de todos los modelos.

Ahora bien, el aprendizaje por arboles de decisión es un método para aproximar funciones objetivo que tomen valores discretos, en los cuales el proceso de aprendizaje se representa por un árbol.

### 4.3.2 Arquitectura e implementación

La arquitectura compuesta de la IA de COVID-X se constituye, por un lado, con un modelo jerárquico de dos CNNs, y por el otro, un modelo basado en *Random Forest*. Cada uno de estos modelos operan con distintos conjuntos de datos, para el modelo de CNN se utilizan las imágenes de radiografía de tórax, y para el segundo modelo, un conjunto de datos clínicos.

La tarea de ambos modelos es obtener una clasificación sobre su respectiva entrada de datos. La clasificación separa aquellos casos en los que se detecta COVID-19. Luego los resultados son operados para obtener un resultado final. En la Figura 25 se puede observar el modelo conjunto, resultante de la combinación del modelo basado en redes neuronales convolucionales llamado **COVID-X-NET**, y el modelo que utiliza el método de *Random Forest* en base a datos clínicos, llamado **COVID-X-CLINICAL**.

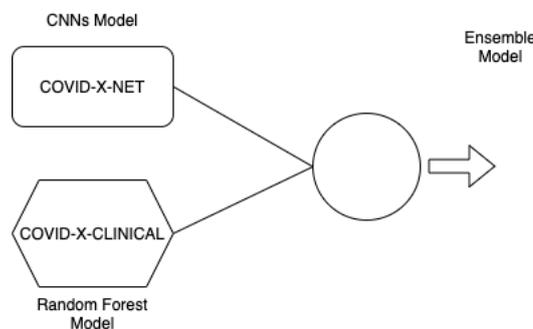


Figura 25: Modelo conjunto.

#### 4.3.2.1 Modelo basado en redes neuronales convolucionales

Para el diseño de este modelo, se toma la decisión de componerlo por un modelo jerárquico de dos CNN debido a que las pruebas realizadas con un solo modelo no dieron buenos resultados. La idea detrás de esta división, es disminuir el alcance de la tarea de cada red. Así es como una se encarga de clasificar si la imagen presenta síntomas de una neumonía o no (llamada CNN-PNEUMONIA), mientras que la otra se especializa en determinar si

dado que hay una neumonía, esta es producto de la enfermedad COVID-19 (llamada CNN-COVID). La predicción final de la imagen se calcula como el producto de analizar el árbol de posibilidades.

La arquitectura de la red encargada de discriminar si se trata de una neumonía o no, se detalla en la Figura 26, y la arquitectura de CNN-COVID se detalla en la Figura 27. Como se aprecia, ambas arquitecturas son similares, esto es debido a que se encontró un buen rendimiento de esta arquitectura con el análisis de imágenes de rayos x. Sin embargo, en un futuro esto podría cambiar, es decir, se podría diseñar una arquitectura mas especializada para cada tarea.

Los bloques llamados *Conv Block* que se muestran en ambas figuras, son composiciones de una secuencia de transformaciones:

1. Convolución separable: *kernel* 3x3 con función de activación *Relu*
2. Convolución separable: *kernel* 3x3 con función de activación *Relu*
3. Normalización en lotes
4. Submuestreo por máximo: tamaño de ventana: 2x2

En si, todas las convoluciones de los modelos de COVID-X usan la función de activación *Relu* para agregar la no linealidad. Las convoluciones separables son introducidas junto a *MobileNets* [14], los investigadores proponen utilizar una nueva arquitectura de convoluciones, conocida como convoluciones por profundidad separables. Una convolución por profundidad separable se define a partir de dos convoluciones clásicas. La idea de estas convoluciones es separar las etapas de filtrado y de combinación de una convolución normal en dos pasos separados. Así se reduce los tamaños de los *kernels* utilizados, y a su vez la cantidad de parámetros a optimizar.

La arquitectura no se diseña muy profunda por la escasa cantidad de imágenes con las que se cuenta inicialmente. Al reducir el tamaño de la red, la cantidad de parámetros a optimizar no es muy grande, y dado que no se poseen tantas imágenes, de esta forma se evita un *overfitting*.

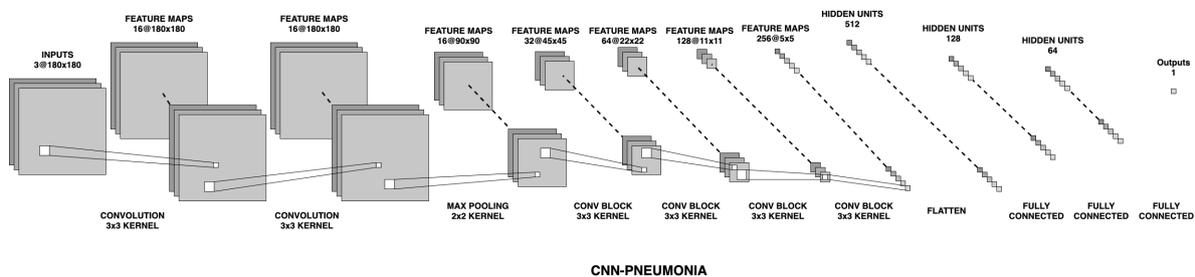


Figura 26: Arquitectura de la CNN-PNEUMONIA.

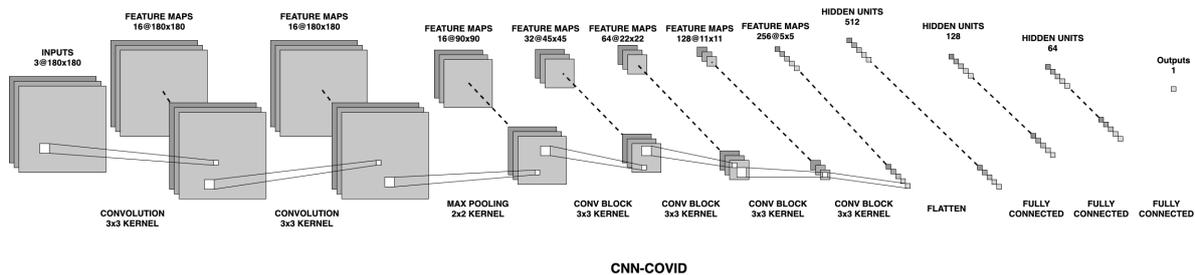


Figura 27: Arquitectura de la CNN-COVID.

Para la implementación de estos modelos de redes neuronales artificiales convolucionales, se utiliza *Keras* sobre la biblioteca de código abierto para aprendizaje automático desarrollada por Google, llamada *TensorFlow* en su versión 2.2.3.

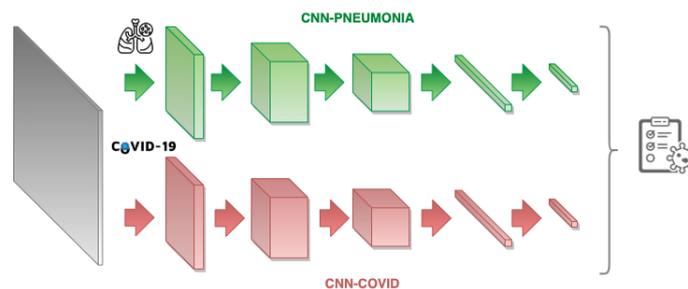


Figura 28: Diagrama de la arquitectura del modelo de composición de CNN-PNEUMONIA y CNN-COVID.

La composición de ambos modelos se muestra en la Figura 28. El resultado será entonces en base a la interpretación de los resultados parciales de ambas redes, siguiendo las reglas de decisión del modelo jerárquico diseñado. De evaluar la salida de CNN-PNEUMONIA, se determina si se detectan signos de neumonía  $P(\text{NEUMONÍA})$  (probabilidad de neumonía). Luego, CNN-COVID determina si hay presente una neumonía viral causada por la enfermedad COVID-19  $P(\text{COVID}|\text{NEUMONÍA})$ . Esta distribución es aprendida debido al entrenamiento con todas imágenes con neumonía. El resultado se conforma por: la probabilidad de no presentar neumonía (obtenida por CNN-PNEUMONIA), y en caso de presentarla, esta probabilidad se multiplica con la obtenida por CNN-COVID, dando si es neumonía que tan probable es que sea causada por la enfermedad COVID-19.

Expresado en términos matemáticos se puede definir la salida final de COVID-X-NET como: probabilidad de no presentar neumonía  $P(\overline{\text{NEUMONÍA}})$  definida en la Ecuación 1, probabilidad de presentar neumonía causada por

COVID-19  $P(\text{COVID} \cap \text{NEUMONÍA})$  definida en la Ecuación 2, y probabilidad de presentar neumonía no causada por COVID-19  $P(\overline{\text{COVID}} \cap \text{NEUMONÍA})$  definida en la Ecuación 3.

$$P(\overline{\text{NEUMONÍA}}) = 1 - P(\text{NEUMONÍA}) \quad (1)$$

$$P(\text{COVID} \cap \text{NEUMONÍA}) = P(\text{COVID} | \text{NEUMONÍA}) * P(\text{NEUMONÍA}) \quad (2)$$

$$P(\overline{\text{COVID}} \cap \text{NEUMONÍA}) = P(\text{NEUMONÍA}) - P(\text{COVID} \cap \text{NEUMONÍA}) \quad (3)$$

#### 4.3.2.2 Modelo basado en random forest

Con el fin de implementar este modelo que hace uso de los datos clínicos provistos del paciente para complementar la clasificación, se utiliza la implementación de *Random Forest* de la librería del lenguaje de programación *Python* llamada *Scikit Learn* donde se implementa esta técnica mediante la clase *RandomForestClassifier*.

En el modelo COVID-X-CLINICAL, en lo que respecta a la función de ganancia utilizada por los algoritmos que construyen árboles de decisión, se utiliza el coeficiente de *Gini*. Al configurar el *RandomForestClassifier*, se indica que emplee *Bootstrap Aggregating*. Luego también se controla tanto la aleatoriedad del *bootstrapping* de las muestras utilizadas para construir árboles, como el muestreo de las características a considerar cuando se busca la mejor división en cada nodo. Para realizar esto último, se configura el parámetro *random state* del clasificador.

#### 4.3.2.3 Composición de ambos modelos

Se utiliza un método para componer los resultados de cada modelo con el fin de obtener un resultado final. Para llevar a cabo esta operación, se hace uso de un método simple, que se basa en un perceptron que toma como argumentos el vector de salida de ambos modelos (vector de probabilidad), y luego los promedia según una suma pesada. Esta técnica es una de las tantas técnicas para construir un *ensemble model*. Este es a su vez entrenado para determinar sus pesos.

### 4.3.3 Conjunto de datos

#### 4.3.3.1 Determinación de datos clínicos complementarios

Se han examinado estudios con modelos de inteligencia artificial basados únicamente en el análisis de imágenes de radiografía de tórax. Muchos de ellos [26], muestran resultados favorables, pero carecen de un respaldo contundente

en los datos. Por estos motivos y en búsqueda de aumentar la confianza de los modelos, se decide integrar el modelo complementario basado no en la radiografía misma, sino que en datos clínicos de la persona.

Estos datos deben ser tales que describan características presentadas por los casos sospechosos de infección. Para determinar este conjunto de datos, partimos de un estudio [5] en los que se tienen en cuenta diversas comorbilidades:

	Característica	Tipo	Detalle
1	Sexo	String	Femenino, masculino.
2	Edad	Entero	El rango de edad es de 0 a 96 años.
3	Fiebre	Booleano	Temperatura mayor a 38 °C.
4	Tos	Booleano	Tos seca o tos con flema.
5	Neumonía	Booleano	Hospitalizado por neumonía.
6	Infección pulmonar	Booleano	Derivado de analizar una CT.
7	Goteo nasal	Booleano	Presenta goteo nasal.
8	Diarrea	Booleano	Hospitalizado con diarrea.
9	Historial de viajes	Booleano	Sospechosos de viajar a una o más destinos.
10	Aislamiento	Booleano	Estado de tratamiento de aislamiento.

Tabla 2: Descripción de características clínicas tomadas como modelo base.

Se consulta con especialistas profesionales de la salud para evaluar la viabilidad de la obtención de estos datos en el contexto de la emergencia sanitaria de nuestro país. A su vez, con ellos se indaga sobre cuales características clínicas consideraban relevantes en el diagnostico de COVID-19. Se realizaron varias iteraciones en la definición de este conjunto de datos en las que intervinieron el departamento de radiología del hospital Gutierrez y nuestro tutor externo, Sergio Herrera. El conjunto de datos resultante de estas iteraciones es el siguiente:

	Característica	Tipo	Detalle
1	Sexo	String	Femenino, masculino u otro.
2	Fecha de nacimiento	RFC3339	El rango de edad de 0 a 120 años.
3	Temperatura corporal	Decimal	Temperatura en °C.
4	Altura	Decimal	En metros.
5	Peso	Decimal	En kilogramos.
6	Asma	String	Si, no, o indefinido.
7	EPOC	String	Si, no, o indefinido.
8	Presión parcial de oxígeno	Decimal	En mmHg.
9	Saturación de oxígeno	Decimal	Porcentaje
10	Frecuencia cardíaca	Entero	En latidos por minuto.
11	Frecuencia respiratoria	Entero	En respiraciones por minuto.
12	Presión arterial	Decimal	En mmHg.

Tabla 3: Descripción de características clínicas determinadas para su recolección por el sistema.

#### 4.3.3.2 Recolección datos

Uno de los objetivos primarios de COVID-X es la recolección de datos clínicos relevantes para confeccionar un conjunto de datos de entrenamiento para los modelos de inteligencia artificial desarrollados dentro de esta plataforma. En una etapa temprana, sin la puesta en marcha de un sistema que permita recolectar esta información, se recurre a conjuntos de datos externos para entrenar los modelos desarrollados.

Este conjuntos de datos inicial se conforma por varios conjuntos externos, detallados a continuación:

1. *CheXpert*
2. Conjunto de datos investigación *Waterloo*
3. *C3 AI COVID-19 Data Lake*
4. covid-chestxray-dataset

El conjuntos de datos consta de imágenes JPEG en las cuales corresponden con radiografías en plano axial y lateral de pacientes diagnosticados con COVID-19, neumonías provenientes de otras causas y pacientes sin afecciones. Los pacientes son de varias locaciones, habiendo casos de Canadá, Estados Unidos, España, Italia, China, Argentina, entre otros.

Este conjunto de imágenes médicas esta compuesto por: imágenes de neumonía obtenidas de *CheXpert* [16], donde se obtienen 2455 imágenes anotadas con indicadores de neumonía viral; imágenes de *C3 AI COVID-19 Data*

*Lake*, la cual contiene 311 imágenes de distintos pacientes; 100 imágenes de COVID-19 positivo que fueron publicadas por el equipo de investigación de Waterloo [26]. Cabe destacar que este mismo equipo publicó, entre sus conjuntos de datos utilizados, un conjunto que cuenta también con anotaciones que describen datos clínicos de los pacientes junto con la radiografía. Este conjunto se conoce como *covid-chestxray-dataset*, y es de vital importancia para poder construir un conjunto de datos para el entrenamiento del modelo compuesto COVID-X-AI.

En total se cuenta con un conjunto de imágenes inicial de 411 imágenes correspondientes a casos de COVID-19 positivos, 2455 de neumonía viral, y con 700 imágenes de radiografías sin ningún hallazgo (también provistas por Chexpert).

Es importante destacar de los casos positivos de COVID-19 son un 55% pacientes de origen Europeo, esto es importante dado que las cepas que se encuentran en nuestro país mantienen una cercanía a aquellas encontradas en el continente europeo.

Por otro lado, el conjunto de datos provenientes de *C3 AI COVID-19 Data Lake* esta compuesto por información clínica. De los datos disponibles se logra generar un subconjunto de información curada según las variables de interés planteadas por el equipo de expertos. Este subconjunto se compone de: edad, sexo, resultado de RT-PCR, temperatura y porcentaje de saturación de oxígeno en sangre.

Luego de eliminar casos con valores nulos, se obtienen 1711 entradas, correspondientes a 211 pacientes. Dichos pacientes recibieron un seguimiento médico lo cual otorga más datos que miden la evolución temporal de la enfermedad. Aunque este último análisis no se tomó en cuenta, es una posibilidad que lo consideremos para futuros estudios.

#### **4.3.4 Entrenamiento**

Para llevar a cabo el entrenamiento de los modelos, se implementa un marco de trabajo llamado COVID-X-TRAINER. A continuación se presentan detalles del mismo, y a su vez del modo de uso, y finalmente se discute la metodología de entrenamiento.

##### **4.3.4.1 COVID-X-TRAINER**

El modulo o componente llamado COVID-X-TRAINER conforma un *framework* controlado para entrenamiento de estimadores. El esquema de la Figura 29 muestra cómo se debe interpretar cada submódulo. Desde una mirada macroscópica, este sistema consta de 3 submódulos funcionales que

son *Data Handler*, *Data Format* y *Model Exploration*, los cuales se encargan de partes específicas del pipeline de entrenamiento.

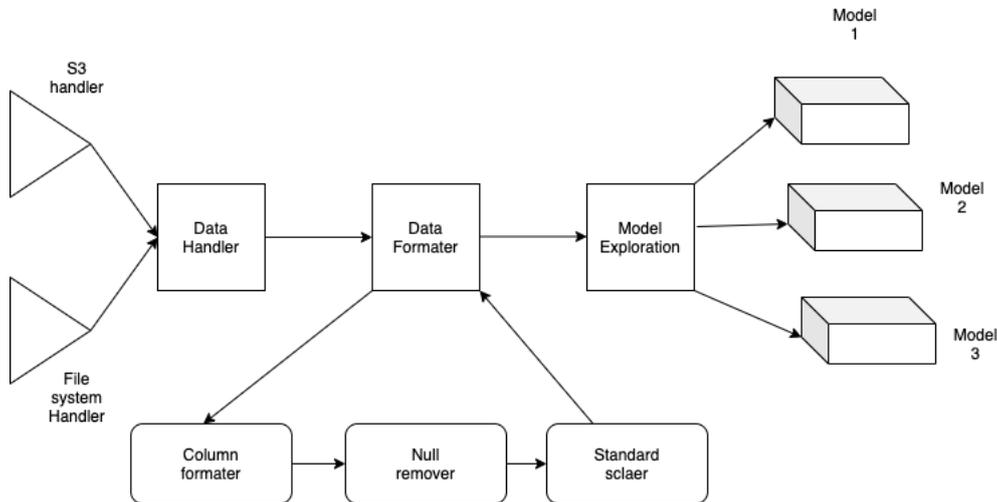


Figura 29: Diagrama de bloques de COVID-X-TRAINER.

- ***Data Handler***: Se encarga de suministrar información que proviene de las posibles instancias de *handlers* específicas. El modelo es extensible siguiendo el patrón de inyección de dependencias mediante un archivo de configuraciones. Se espera que cada instancia de *handler* provea una implementación de la interfaz planteada con la cual se ponen a disposición los datos del *source* indicado. En particular, esto permite tener *handlers* para consumir tanto los conjuntos de datos externos, así como también una instancia que actúe como adaptador para consumir los estudios recolectados por COVID-X-API.
- ***Data Format***: Se trata del submódulo encargado de aplicar el ciclo de preprocesamiento sobre los datos que irán al modelo. Dado que esta arquitectura fue pensada para un entrenamiento *"out of core"* en modo *batch*, las instancias que se desean utilizar deben estar diseñadas para *streaming processing* ya que no se garantiza tener acceso a la totalidad del conjunto de datos.
- ***Model Exploration***: Este submódulo actúa como administrador para el entrenamiento de múltiples modelos en paralelo utilizando el mismo conjunto de datos. Esto es útil para la reproducibilidad de los resultados, ya que permite eliminar una variable del proceso para la comparación de modelos y su poder de generalización. Cada instancia de *Model*

debe cumplir con la interfaz provista para asegurar que las métricas adjuntas del modelo se guarden.

#### 4.3.4.2 Metodología de entrenamiento

Para el entrenamiento se hace uso del módulo descrito en la sección anterior, además, se implementa un *Data Handler* adecuado para consumir el conjunto de datos externo alojado en AWS. Para cada modelo, se utilizaron conjuntos de datos acordes a los mismos, según la tarea a realizar de cada modelo. Se obtienen tres conjuntos de datos:

1. *Dataset pneumonia*: Esta constituido por imágenes de radiografía de tórax, puede o no presentar neumonía. Este es usado para el entrenamiento del modelo CNN-PNEUMONIA.
2. *Dataset covid*: Utilizado por CNN-COVID, esta conformado por imágenes de radiografía de tórax que presenten neumonía y algunas de ellas causadas por COVID-19.
3. *Dataset clinical data*: Constituido por datos clínicos de personas que poseen o no la enfermedad COVID-19. Es usado para el entrenamiento de COVID-X-CLINICAL.
4. *Dataset image and clinical data*: Este es el conjunto de datos más complejo. Contiene imágenes de radiografía de tórax con sus respectivos datos clínicos, puede presentar neumonía causada por COVID-19. Este conjunto de datos es el utilizado para el entrenamiento de COVID-X-AI.

Se requiere un *Data Handler* diferente para consumir los datos recolectados por el sistema, pero este conjunto de datos debe ser significativamente grande para aplicarse al entrenamiento.

Para la serialización de los modelos, se desarrolla la librería *adhoc COV-ESTIMATOR*, dado que muchos de los modelos utilizan múltiples librerías que no tienen una simple interfaz de integración. Esta librería permite trabajar de forma declarativa con un grafo computacional en el cual cada nodo representa una unidad de computación que puede ser:

- Una red neuronal artificial implementada con *Keras*.
- Un modelo predictor que utilice el estándar de *Scikit-Learn*.
- Una función anónima que especifique un cómputo elemento por elemento.

De esta manera, se obtiene una operatoria de trabajo unificada para todos los modelos, siempre que puedan ser representados como una composición de operaciones o estimadores. El proceso de entrenamiento puede demorar mucho tiempo, por lo que se emplean instancias de *AWS* optimizadas con poder de operaciones *CUDA* y un ancho de banda significativo.

Este proceso se dispara de forma manual, aunque la serialización del modelo en un bucket de *AWS S3* es automática para optimizar el tiempo de análisis entre modelos. Todo modelo, se aloja en un *AWS S3* al cual nos referimos como *Model Zoo* en donde se encuentran los *artifacts* definidos con COV-ESTIMATOR. Estos modelos tiene una convención de nombres con la cual el modelo que se utiliza en producción al cual llamamos COVID-X-AI, siempre debe guardarse bajo la ruta "*prod*".

#### 4.3.5 Metodología de evaluación de modelos

Una vez que un modelo se entrena y su serialización se lleva al *Model Zoo*, se analizan las métricas de evaluación y sus gráficos de resultados para el conjunto de datos de evaluación (conjunto de datos que se encuentra separado de la información de entrenamiento y de test). Esto se analiza manualmente para así finalizar el proceso de "*blessing*", y si el encargado determina que el modelo es apto para reemplazar al modelo actual de producción, se sube a la ruta correcta.

La razón para mantener esta etapa sin automatización, está fundada sobre la prevención de "**concept drift**". Dado que la información disponible crece y se modifica momento a momento, sumado a que no hay un concepto establecido sobre como se presenta la neumonía causada por COVID-19, existe la posibilidad que nuevas entradas en el conjunto de datos de entrenamiento generen un *concept drift* sobre cómo funciona el modelo.

Para llevar a cabo la evaluación de los modelos, se reserva parte del conjunto de datos correspondiente a cada uno, que sera utilizado para realizar las pruebas y obtener métricas de ellos. Luego así, es posible realizar comparaciones respecto a ellos y definir cual es mas adecuado. Para dividir el conjunto de datos disponible en un conjunto de entrenamiento y otro como conjunto de pruebas se utiliza la función *train\_test\_split* de *sklearn.model\_selection* que estratifica las clases en la división.

Las métricas de evaluación de los modelos de inteligencia artificial implementados incluyen: matriz de confusión, sensibilidad, precisión, exactitud (*accuracy*), *F1 Score*, error estimado del método de clasificación, error estimado relativo del método de clasificación, curvas *ROC* y el área bajo las curvas o *AUC*.

La métrica *AUC* refleja la probabilidad de que un modelo de reconocimiento

clasifique un caso positivo elegido al azar más alto que un caso negativo elegido al azar. Es la métrica más utilizada para evaluar el rendimiento general de la clasificación. La sensibilidad da la proporción de positivos identificados correctamente. La precisión da el porcentaje de casos clasificados correctamente, incluidos los positivos y negativos. La curva ROC, muy usada en el ámbito médico como método estadístico para determinar la exactitud diagnóstica de pruebas, da una representación gráfica de la sensibilidad frente a la especificidad.

## 4.4 Backend

El backend de COVID-X está conformado por un conjunto de APIs REST, basando la arquitectura de microservicios. Para maximizar el rendimiento, la arquitectura se piensa de manera que los componentes sometidos a tensión se puedan escalar horizontalmente, lo que reduce los tiempos de respuesta y las tasas de error durante las cargas máximas.

A continuación se detallan las API REST de COVID-X, la documentación de los principales endpoints expuestos por estas se encuentra en el Anexo A.

### 4.4.1 DICOMVERTER

La aplicación DICOMVERTER es una API REST que constituye un microservicio dentro de la arquitectura de COVID-X. Este se encarga del procesamiento, conversión y almacenamiento de imágenes DICOM. Las imágenes médicas por sí solas no aportan suficiente información, para interpretarlas correctamente es necesario que se complementen con datos del paciente. Los formatos tradicionales como JPEG, PNG o BMP no permiten incluir estos datos extra. El formato DICOM permite tener no solo la imagen médica sino que también toda esta información o metadatos adicionales del paciente y del análisis clínico.

El flujo de este servicio estima que los usuarios de COVID-X proporcionen al sistema imágenes DICOM, luego, DICOMVERTER procesa estas imágenes, extrae información de los metadatos, y convierte la imagen médica en formato BMP.

Para llevar adelante esta funcionalidad, se opta por utilizar el lenguaje de programación *Java* en su versión 8, junto con *Maven* como gestor de dependencias, y *Spark* como enrutador HTTP de alto rendimiento. La elección de este lenguaje resulta ideal para hacer uso de las librerías médicas para manejo de todas las acciones y formatos relacionados con el protocolo DICOM llamada *DCM4CHE*. Esta es una librería de código abierto muy madura, y con una gran comunidad de apoyo y documentación específica.

El kit de herramientas y la biblioteca *dcm4che DICOM* se utiliza en muchas aplicaciones de producción en todo el mundo y la versión actual (5.x) se ha rediseñado para ofrecer un alto rendimiento y flexibilidad. Para la implementación de DICOMVERTER se emplea la versión 5.14.1, además de los utilitarios que permiten procesar y acceder a los metadatos de la imagen DICOM, el modulo de conversión para convertir la imagen médica a mapa de bits, y finalmente también son utilizadas las funcionalidades provistas para comunicarse con un servidor PACS.

Un servidor PACS es un sistema de almacenamiento digital, transmisión y descarga de imágenes radiológicas. DICOMVERTER envía los archivos DICOM procesados para ser almacenados en un COVID-X-PACS, que se trata de un software de código abierto llamado *Orthanc*, un servidor DICOM independiente simple pero potente.

El servicio expuesto, comienza procesando los metadatos, accediendo a la etiqueta (0020,000D) para obtener la *StudyInstanceUID*. Este identificador es un identificador global único de todas las radiografías en formato DICOM. Luego se convierte la imagen médica a mapa de bits y finalmente, se envía el fichero original al servidor PACS. El servidor PACS internamente utiliza una base de datos relacional, y de esta manera, provee herramientas de búsqueda que permite tener mayor control y potencia de búsqueda sobre todos los ficheros DICOM procesados por nuestro ecosistema. El servicio descripto, luego de completar la secuencia detallada anteriormente, retorna una imagen BMP y el *StudyInstanceUID*.

#### 4.4.2 COVID-X-API

La API llamada COVID-X-API es la principal en el *backend* de COVID-X. Entre sus responsabilidades se encuentra el manejo de la lógica de negocio donde se modelan los flujos del sistema tanto para carga de datos de entrenamiento como para realizar inferencias a la inteligencia artificial. Interactúa con servicios propios y de terceros, así como también con la base de datos de estudios.

Para la implementación de esta API se hace uso del lenguaje de programación *Golang*. Es un lenguaje de programación creado por Google en 2009, también conocido como *Go*. Lo que caracteriza a este lenguaje es que es compilado, concurrente, imperativo, estructurado, orientado a objetos, inspirado en la sintaxis del lenguaje C, y tipado. Su diseño permite que los programadores del lenguaje C migraran a un lenguaje actual pero con la misma potencia. Ha ganado mucha fuerza, empresas de renombre confían en *Golang*, entre ellas Docker, Facebook, Netflix, Uber, Twitter, y Mercadolibre.

Los creadores de *Golang*, en especial sus diseñadores iniciales Robert

Griesemer, Rob Pike y Ken Thompson, se propusieron crear un lenguaje fácil de aprender, sencillo de codificar y robusto. Esto lleva a que *Go* sea muy eficiente en el procesamiento de datos a gran escala y aplicaciones que manejan grandes cantidades de datos. Además de sus múltiples características, cuenta con una destacable implementación de su *garbagecollector*, y en lo que respecta a su compilador, facilita la *crosscompilation* resultando en una compilación multiplataforma.

Una vez definido el lenguaje de programación a utilizar, para garantizar un alto rendimiento y tiempos de respuesta reducidos en los flujos principales del sistema, se opta por el enfoque de diseño guiado por el dominio o *DDD* (*Domain Driven Design*) para el desarrollo del software.

Se comienza con la definición de un lenguaje ubicuo, común y riguroso entre desarrolladores y usuarios. Este lenguaje debe basarse en el modelo de dominio que se utiliza en el software, de ahí la necesidad de que sea riguroso, ya que el software no se adapta bien a la ambigüedad.

Como sostiene Eric Evans en [11], al utilizar el lenguaje basado en modelos de forma generalizada y no quedar satisfecho hasta que fluye, se acerca un modelo que es completo y comprensible, compuesto por elementos simples que se combinan para expresar ideas complejas. El uso de este lenguaje permite poder tomar decisiones en relación a cuestiones médicas con el apoyo de los colaboradores provenientes del área médica.

Se dispone una arquitectura de capas, contando con 4 capas para la implementación de COVID-X-API, como se puede observar en la Figura 30. La primer capa *INTERFACES* se encarga de la interacción con los usuarios o clientes, tanto presentado y recibiendo información del usuario. Esta capa dispone en su implementación los controladores de la API REST y los objetos de transferencia de datos o *DTO* (Data Transfer Object por sus siglas en inglés). La segunda capa *APPLICATION* es un intermediario entre la interfaz y el dominio. La capa de *DOMAIN* es el centro del software, contiene la lógica del dominio y el conocimiento del negocio. En la capa de dominio existen componentes que pueden ser entidades, objetos que tienen una identidad, objetos de valor donde solo interesan sus atributos y no la identidad, objetos de acceso de datos y servicios que implementan la lógica de negocio. Por último, la capa de *INFRASTRUCTURE* es una capa de soporte para las otras tres capas, contiene bibliotecas de soporte, utilitarios y servicios externos.

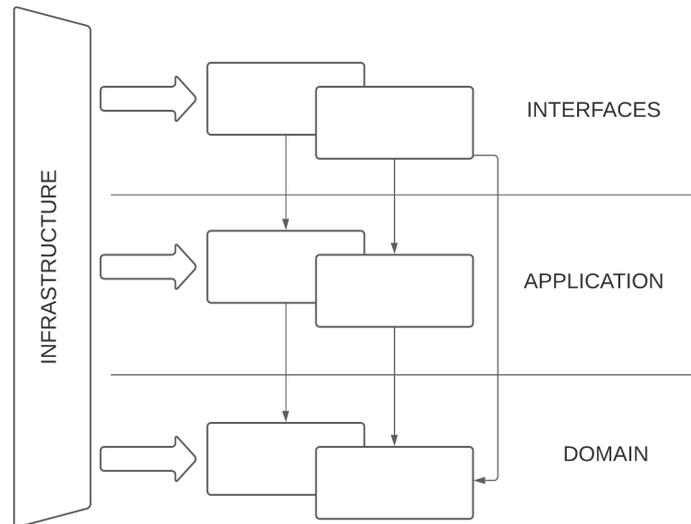


Figura 30: Diagrama en bloques de las capas en la arquitectura de COVID-X-API

Dentro de la capa de dominio se encuentra la entidad de *Study* que representa un estudio clínico en el sistema y las inferencias realizadas sobre el mismo, también se definen sus objetos de acceso de datos y servicios. En cuanto al tipo de persistencia de dichos estudios e inferencias se utiliza una base de datos no relacional para almacenar los datos clínicos no estructurados. Basados en el teorema CAP [13], se opta por utilizar *MongoDB* para el almacenamiento de estos datos. Considerando el teorema CAP, *MongoDB* es una base de datos *CP* (consistencia y tolerancia a particiones), es decir, esta resuelve las particiones de red manteniendo la consistencia comprometiendo la disponibilidad, dado que como afirma el teorema no se pueden obtener las tres cualidades. En lo que respecta a las imágenes BMP de los estudios, las mismas se almacenan en un contenedor de archivos, en particular en servicio provisto por *Amazon Web Services* llamado *Simple Storage Service* o *S3*.

El proyecto *Golang*, se desarrolla en su versión 1.15, y *go modules* como manejador de dependencias. Para estructurar el proyecto se siguen los estándares impuestos por la comunidad de *Golang*, y dentro de esa estructura se montan las capas descritas anteriormente, teniendo en cuenta el uso de patrones de diseño de software en su implementación. Por otra parte, se utiliza un framework web llamado *gin gonic* o también llamado *Gin Web Framework*, como enrutador o multiplexador HTTP de la API. Este provee un alto rendimiento y tiempos de respuesta muy bajos aportando al desempeño global del sistema.

Por ultimo, cabe mencionar que se utiliza una herramienta de análisis estático de código para el desarrollo del servicio, haciendo uso de diferentes linters incluyendo *golangci-lint* configurado restrictivamente para seguir los lineamientos definidos por el equipo de desarrollo. En lo que respecta a las pruebas, se implementan pruebas unitarias de los componentes desarrollados así como también se añadieron pruebas de integración para los flujos principales. Para ambos tipos de pruebas, se hace uso las herramientas de apoyo provistas por *testify*.

#### 4.4.3 COVID-X-USERS

En lo que respecta a COVID-X-USERS, originalmente se desarrolló acoplada en COVID-X-API, pero con el transcurso del proyecto se separó del mismo, en un microservicio de usuarios de COVID-X.

Todas las consideraciones de diseño que aplican a COVID-X-API, también lo hacen para COVID-X-USERS, es otra API REST Golang con los patrones y arquitectura idénticos, haciendo uso del enfoque de diseño guiado por el dominio en el desarrollo.

Dentro de su dominio, esta presente la entidad de usuario y en lo que respecta a su persistencia, se utiliza una base datos relacional *PostgreSQL* mediante la librería de Golang de asignación objeto-relacional (más conocido como Object-Relational Mapping u ORM por sus siglas en ingles) llamada *GORM*. Esto permite realizar un mapeo directo del modelo o entidad a la relaciones y entidades que maneja el paradigma de las bases de datos relacionales.

La información de los usuarios y organizaciones se almacena como datos estructurados con un esquema y ciertas restricciones impuestas por las entidades y relaciones definidas. Este tipo de base de datos es transaccional, se entiende por transacciones a un conjunto de instrucciones para trabajar con la base de datos. Éstas, según el estándar de las bases de datos relacionales, deben cumplir con las propiedades de atomicidad, consistencia, aislamiento y durabilidad (ACID por sus siglas en ingles). Este carácter transaccional nos permite mantener la consistencia e integridad de los datos de los usuarios y organizaciones de COVID-X.

Para lograr una mejor separación de responsabilidades entre los distintos usuarios al mismo tiempo que una mejor trazabilidad y seguridad, se discriminan a estos siguiendo el principio de menor privilegio. Este principio se basa en que cada usuario o parte del sistema debe ser capaz de acceder solo a la información y recursos que son necesarios para su legítimo propósito. De esta manera, se diseñan roles para los usuarios de COVID-X:

- COVID-X-TRAINING-SET-WRITER: Los usuarios que posean este rol pueden realizar cargas de datos para entrenamientos.
- COVID-X-INFERENCE-SET-WRITER: Este rol permite realizar inferencias a los usuarios.
- COVID-X-STUDY-READER: Tiene la finalidad de permitir realizar búsquedas sobre los estudios registrados en el sistema.
- COVID-X-ADMIN: Este ultimo es de moderación para la validación y activación de usuarios.

Finalmente, COVID-X-USERS se construye también usando Golang 1.15 y *go modules* como manejador de dependencias y Gin Web Framework.

#### 4.4.4 INFERENCE-GATEWAY

La aplicación INFERENCE-GATEWAY, es un microservicio dentro del ecosistema de COVID-X, desarrollado con una API REST. Su principal función es exponer un servicio que, dado un nuevo estudio, este realiza una inferencia con el modulo de inteligencia artificial y retorna los resultados de la perdición.

Este es el intermediario entre el resto de los servicios y la inteligencia artificial, permitiendo desacoplar los flujos de inferencia de la implementación de la IA.

Para desarrollar esta API se emplea el lenguaje de programación *python* en su versión 3.8, junto con el framework flask por su simpleza y agilidad para el desarrollo. Expone dos servicios, uno para consultar la ultima versión del modelo de la IA, y otro para realizar una inferencia usando el modelo más reciente, requiriendo ser provisto de los datos clínicos en formato JSON y una URL a la imagen BMP de la radiografía de tórax.

## 4.5 Frontend

El *frontend* de COVID-X se conforma por una aplicación web llamada COVID-X-FRONT, que representa la puerta de entrada del usuario al sistema. Para el diseño de la misma se utiliza un enfoque de diseño centrado en el usuario, junto con técnicas de interacción humano-computador (HCI por sus siglas en ingles) para garantizar la usabilidad de la aplicación web. Como se mencionó en la sección de arquitectura, la usabilidad es un atributo de calidad que se tiene en consideración para el diseño de la misma.

### 4.5.1 Usuarios de COVID-X

Dado que el usuario promedio esperado del sistema es un usuario miembro de una institución médica, personal de la salud, se tiene en cuenta la retroalimentación de los usuarios tempranos de la plataforma para iterar la interfaz de usuario en búsqueda de mejor experiencia de usuario.

Como resultado del análisis de operación del sistema, se observa que el tipo de usuarios, por lo general son usuarios no técnicos en lo que respecta a la informática, y a su vez son especializados en el ambiente médico, el contenido del sitio web debe estar alineado con su modelo mental. Teniendo especial atención en el correcto uso de términos médicos específicos, evitando cualquier tipo de ambigüedad. Una vez definido el perfil de las personas que representan el usuario medio del sistema, se llevan a cabo pruebas de usabilidad con los diseños preliminares del sitio web, evaluando el diseño con personal médico voluntario.

Por otra parte, se estima que los usuarios estén atendiendo la presente emergencia sanitaria durante el uso del sistema, con lo cual, sus tiempos se verán reducidos para interactuar con este. Por estos motivos, para el diseño de la interfaz se tiene muy en cuenta que esta sea rápida, de bajos tiempos de respuesta, e intuitiva. Esto se evalúa en las pruebas de usabilidad, y además se busca en las iteraciones del diseño reducir la métrica de cantidad de clicks y la cantidad de flujos concretados con éxito. El objetivo es lograr que los usuarios de COVID-X tengan una buena experiencia al utilizar la interfaz, sintiendo que tienen control sobre la misma.

### 4.5.2 COVID-X-FRONT

La aplicación COVID-X-FRONT, es una aplicación web en una sola página (SPA por sus siglas en ingles), es decir, una página web en la cual se presenta todo el contenido en una sola página.

Para el desarrollo de la misma se emplea la biblioteca del lenguaje de programación *Javascript* llamada *React*. Se trata de una biblioteca de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de este tipo de aplicaciones. Es mantenida por Facebook y la comunidad de software libre, contando con una licencia MIT.

En la aplicación COVID-X-FRONT, se utiliza la versión 16.13.1 de la biblioteca *React*. Resumiendo, *React* representa la vista en un contexto en el que se usa el patrón MVC (Modelo-Vista-Controlador). Esta basado en componentes, estos se crean encapsulados y manejan su propio estado, además, se componen para formar parte de una interfaz de usuario compleja. Posee un carácter declarativo, diseñando vistas simples para cada estado de

la aplicación, y se encarga de actualizar y renderizar de manera eficiente los componentes correctos cuando los datos cambian.

Las vistas declarativas hacen que el código sea más predecible, por lo tanto, más fácil de depurar. La aplicación hace uso de las ventajas de React en conjunto con nuevas APIs de las recientes versiones de la biblioteca. La API de Context de React (disponible desde la versión 16.3) permite encapsular parte del estado en un contexto que es inyectable en cualquier lugar del árbol de componentes. Por otra parte, la API de Hooks (disponible desde la versión 16.8) permite usar estado y otras características de React sin la necesidad de escribir una clase. Los Hooks son funciones que te permiten enlazar el estado de React y el ciclo de vida desde componentes funcionales.

La disposición de la pagina web es el cuerpo central en el medio de esta, y una barra lateral de navegación con las distintas pestañas o secciones del sitio. Al entrar en el sitio, se muestra la página de inicio, donde se expone contenido con una breve introducción al sistema COVID-X. Luego los usuarios que deseen usarlo pueden ver fácilmente un botón que los llevara a iniciar sesión para comenzar su uso.

Entre las pestañas disponibles en la barra lateral, se encuentran: un *blog* para discusiones de temas asociados, una pagina con fundamentos, documentos y legales relevantes, una pestaña que referencia a la pagina de inicio, y por ultimo, una para hacer inicio de sesión o registro de usuario.

Al iniciar sesión exitosamente, una nueva pestaña es introducida al menú lateral que tiene la finalidad de llevar a cabo una inferencia o una carga de imagen para entrenamiento. Dependiendo del rol del usuario, se visualiza una de ellas o ambas, para realizar cualquiera, el usuario debe cargar la imagen de radiografía de tórax, y completar los datos clínicos.

Debido a la potencial demanda de una organización internacional ubicada en Estados Unidos, se decide priorizar la internacionalización del sitio web haciendo uso de una de las practicas comunes llamada *i18n*. Esta permite rápidamente adaptar el sitio web y el contenido de los correos electrónicos que reciben los usuarios, tanto al idioma ingles como español. De esta manera, se amplia el alcance del proyecto, posibilitando a que entidades extranjeras puedan utilizar el sistema.

Para mas información sobre la presentación de la pagina web, se puede consultar la sección Manual de uso.

## 4.6 Infraestructura

La arquitectura del sistema COVID-X se encuentra montada sobre la nube. El proveedor de nube publica utilizado es *Amazon Web Services (AWS)*.

En la Tabla 4, se puede observar un listado aproximado de los recursos de *AWS* utilizados para la creación, desarrollo y puesta en producción de la plataforma COVID-X. Como se puede inferir del diagrama de despliegue presentando en la vista física de la arquitectura del sistema en 4.2.2.4, la cantidad de servicios y recursos es alta para ser administrados de forma manual. En los inicios del proyecto se comenzó montando y administrado los recursos de la nube manualmente. Luego, con el transcurso del desarrollo, la cantidad de recursos aumento considerablemente y se evaluó implementar infraestructura como código (IaC por sus siglas en ingles).

IaC es el proceso de gestión, aprovisionamiento, y administración de infraestructura de centro de cómputos a través de archivos de definición. En particular, la infraestructura a manejar es la del proveedor de nube. Los archivos de definición incorporan un carácter o enfoque declarativo para manejar la infraestructura por sobre procesos manuales.

Si bien los recursos no son de tal magnitud que resulten imposibles de manejar de forma manual, se encuentran otras ventajas de implementar un proceso de IaC. Entre ellas, la ejecución es más rápida en lo que refiere a la automatización de la infraestructura y su configuración. Por otro lado, se elimina el riesgo asociado a los errores humanos, derivados de la mala configuración manual. Esto puede disminuir el tiempo y la magnitud de la afectación de los errores en el sistema, y también, ayuda a evitar violaciones de seguridad. Estas ventajas, se encuentran fuertemente alineadas con los atributos de calidad priorizados en el diseño de la solución y altamente compatible con el tipo de desarrollo ágil que se llevó a acabo.

Para implementar IaC, se utiliza el software de infraestructura como código llamado *Terraform*. Este permite definir y configurar la infraestructura en un lenguaje de alto nivel, generando un plan de ejecución para desplegar la infraestructura en proveedores de servicio de nube tales como *AWS*, *IBM Cloud*, *Google Cloud Platform*, *Microsoft Azure*, etc. La infraestructura se define utilizando la sintaxis de configuración *HashiCorp Configuration Language (HCL)* creada por la empresa que desarrolla *Terraform*, *HashiCorp*. Usar esta herramienta que soporta varios proveedores permite migrar de proveedor de nube o desplegar la arquitectura en un esquema de múltiples proveedores de nube publica.

La implementación *Terraform* para la infraestructura de COVID-X, permite crear fácilmente entornos similares productivos, pre productivos y de

pruebas. El entorno pre productivo posibilita probar los cambios incrementales incorporados en el sistema de forma manual para emular un ambiente lo más parecido al entorno de producción, y de esta manera, poder efectuar pruebas manuales antes de desplegar cambios en producción.

Grupo	Recurso	Cantidad	Notas
VPC	VPC	1	
VPC	Internet Gateway	1	
VPC	NAT Gateway	1	
VPC	Elastic IP	<10	
VPC	Subnet	20	
Route53	Zone	1	
Route53	DNS record	20	
ACM	SSL Certificate	1	
EC2	t2.small	1	Bastion host
EC2	t2.medium	1	Servidor PACS
EC2	Snapshots	5	
EC2	Volumes	5	<1TB
ELB	Application Load Balancer	5	
ELB	Network Load Balancer	1	
ECR	Docker Registry	3	
ECR	Storage	<100 GB-mo	
ECS	Fargate cluster	3	<10 vCPU, 10GB RAM
KMS	Key	<10	
S3	Bucket	10	2 TB
CloudWatch	Metric	50	
CloudWatch	Log event	1GB	
CloudWatch	Log storage	1GB-mo	
Amplify	Storage	<1GB	
CodePipeline	Pipeline	3	
CodeBuild	Monthly build minutes	100	
RDS	db.t2.small	1	<100 GB

Tabla 4: Catalogo con estimación recursos utilizados en *AWS* para el montaje, desarrollo y puesta en funcionamiento del sistema COVID-X.

En resumen, dentro de los recursos utilizados en *AWS*, por un lado se encuentran los relacionados con la persistencia de datos, donde se incluyen *AWS S3 (Simple Storage Service)* para almacenamiento de objetos y *AWS RDS (Relational Database Service)* para base de datos relacionales, así como también una instancia *AWS EC2 (Elastic Compute Cloud)* con capacidad de almacenamiento acorde para el servidor PACS. Por otro lado, los relaciona-

dos con computo, *EC2* para el *bastion host* que se encuentra disponible 24x7, siendo un componente dentro de la cadena de seguridad de las redes virtuales internas de la nube de *AWS* para acceder a los recursos privados del sistema. También los servicios *AWS ECR (Elastic Container Registry)* y *AWS ECS (Elastic Container Service)* con los *clusters* de *AWS Fargate* que permiten orquestar los servicios del *backend* en contenedores de *Docker*. Por último, podemos agrupar otros servicios como *AWS Route53* para el *DNS*, *VPC (Virtual Private Cloud)*, Subredes dentro de las *VPC*, Internet y NAT Gateways, *AWS KMS (Key Management Service)* para manejo de claves criptográficas, *AWS Amplify* para alojar el *frontend*, y por último, *AWS CodePipeline* y *AWS CodeBuild* para automatizar los procesos de despliegue.

El proyecto COVID-X utiliza la plataforma *GitHub*, donde se alojan los proyectos asociados a los módulos que componen la arquitectura del sistema, utilizando el sistema de control de versiones *Git*. Dentro de *GitHub*, existe la organización llamada COVID-X donde se agrupan todos estos proyectos, en los cuales se configura sus repositorios para implementar integración continua y despliegue continuo (CI/CD por sus siglas en inglés).

Con el fin de implementar CI/CD, se utiliza *AWS CodePipeline* en conjunto con *AWS CodeBuild* para automatizar chequeos en los desarrollos sobre los repositorios y el despliegue de nuevas versiones. Al realizar un *merge* en la rama por defecto *master*, se dispara un proceso automático que se encarga de ejecutar test de integración y la construcción, empaquetado, y despliegue de los cambios en el ambiente de producción. De igual forma, al realizar un *merge* sobre la rama *staging* se realiza lo mismo pero sobre un entorno pre productivo.

## 5 Desarrollo

Para llevar a cabo la implementación, puesta en funcionamiento, y toma de decisiones de la arquitectura descrita anteriormente, se opta por una metodología de desarrollo ágil de software. Dentro de los métodos de ingeniería del software basados en el desarrollo iterativo e incremental existentes, se emplea el marco de trabajo SCRUM con *sprints* de 2 semanas de duración y con una planificación de alto nivel con 2 etapas de desarrollo.

Con el fin de tener una mejor visibilidad de las tareas que se llevaban a cabo, la relación entre ellas, y la asignación de las mismas a los miembros del equipo, se decide utilizar un software de administración de proyectos con interfaz web y *mobile* llamado *Trello*. Si bien, este ofrece un tablero *KanBan*, es posible modificarlo según como se definan sus distintas columnas. Esta herramienta, ha permitido organizar la planificación y mantener el estado de forma iterativa y creciente con el transcurso de los *sprints*.

En la definición del tablero utilizado, se destacan 5 principales columnas que describen el flujo de trabajo de desarrollo. Una primer columna llamada *TODO* que representa el *backlog*, donde se encuentran las tarjetas correspondientes a las tareas que se tienen que realizar. Estas pueden estar clasificadas según corresponda (*backend*, *frontend*, IA o infraestructura), también se encuentran ordenadas según su prioridad previamente establecida. Estas tarjetas son asignadas a los miembros del equipo en la planificación de cada sprint.

Por otra parte, se cuenta con la columna *DOING*, bajo ella se encuentran las tareas cuyo desarrollo esta en progreso.

En caso de aplicar, se cuenta con una tercera columna llamada *CODE-REVIEW*. Esta columna abarca las tareas que se han realizado pero requieren revisión de otros miembros para poder realizar la transición a un siguiente estado en el flujo de trabajo. El desarrollo se rige en los valores de programación extrema o XP (por sus siglas en ingles) [1]: simplicidad, comunicación, retroalimentación y coraje.

Un punto crucial para concluir la definición del flujo de trabajo del equipo de desarrollo, es la definición de tarea completada. Las tareas completadas cuentan con un columna llamada *DONE*. Se establece que una tarea esta completada cuando esta desplegada en los entornos productivos de COVID-X e impactando sobre la experiencia y el valor aportado a los usuarios del sistema.

Habiendo definido esto, se introduce un estado previo a *DONE* llamado *READY-TO-DEPLOY*, que agrupa las tareas completadas, probadas, revisadas por otros miembros, y que están listas para ser desplegadas a pro-

ducción. Como se mencionó, el proceso de despliegue esta completamente automatizado e integrado con la plataforma *GitHub*, donde se alojan los proyectos utilizando el sistema de control de versiones *Git*.

## 6 Resultados

En esta sección se presentan los resultados de la puesta en práctica de la metodología de evaluación de los modelos de inteligencia artificial usando los conjuntos de datos detallados. Además, se analizan los tipos de usuarios e integraciones involucradas con el sistema COVID-X desde su puesta en producción.

### 6.1 Evaluación de los modelos de inteligencia artificial

A continuación, se muestran y analizan los resultados obtenidos de las métricas de evaluación introducidas en la sección 4.3.5, para los modelos COVID-X-NET, COVID-X-CLINICAL, y el modelo resultante COVID-X-AI.

Para cada uno de los modelos de clasificación se construye una matriz de confusión que indica el desempeño de los modelos, permitiendo visualizar los verdaderos negativos, falsos negativos, falsos positivos y verdaderos positivos. También, se confecciona la respectiva curva de característica operativa del receptor, *ROC* por sus siglas en inglés, que aporta una representación gráfica de la tasa de verdaderos positivos frente a la tasa de falsos positivos para un clasificador binario. Como en nuestro caso es utilizada para clasificadores multi clase, se arman tomando cada clase contra las demás.

Todos los modelos evaluados dan como resultado una clasificación según las clases:

- COVID-19 (**Y**): Presenta neumonía producto de COVID-19.
- Normal (**N**): No presenta neumonía.
- Pneumonia (**P**): Presenta neumonía producto de otros causantes.

#### 6.1.0.1 Evaluación de COVID-X-NET

La evaluación de este modelo compuesto de redes neuronales artificiales convolucionales, se detalla en la Tabla 5.

El modelo COVID-X-NET se entrenó utilizando en conjunto los datos formados por radiografías de *ChestPert* e imágenes extraídas de la plataforma *C3*, constando aproximadamente con 3000 imágenes. Cada una de las redes que componen este modelo fueron entrenadas con los *datasets* mencionados en la Sección 4.3.4.2 de entrenamiento para CNN-PNEUMONIA y CNN-COVID.

Como este caso de uso prioriza la correcta detección de casos positivos de COVID-19, es posible ver que existe un *trade off* entre la detección de casos sin hallazgos en la imagen y aquellos que presentan síntomas de neumonía. Como se muestra en la Figura 32, los desplazamientos de la curva ROC tanto para el caso de la clase *Normal* como de la clase *Pneumonia*, presentan notables diferencias con respecto a la clasificación de COVID-19. Esto mismo se puede confirmar con la precisión multiclase de la Tabla 5 que se corresponde con la matriz de confusión de la Figura 31.

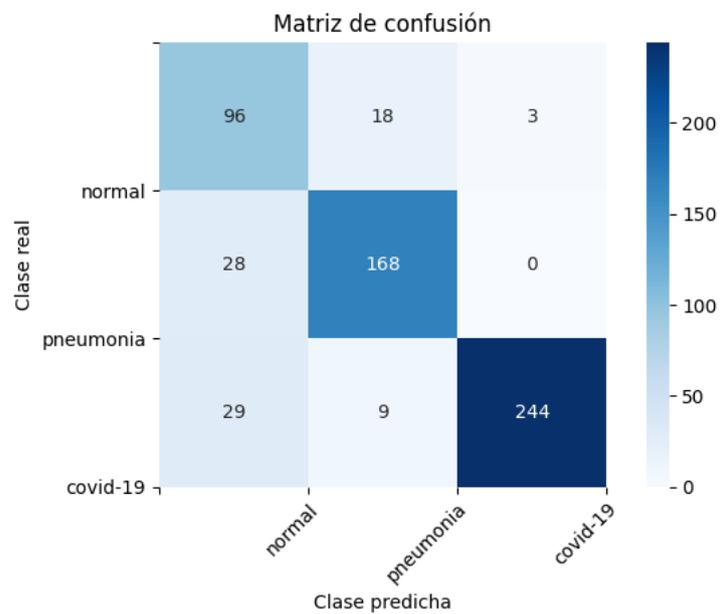


Figura 31: Matriz de confusión del modelo COVID-X-NET

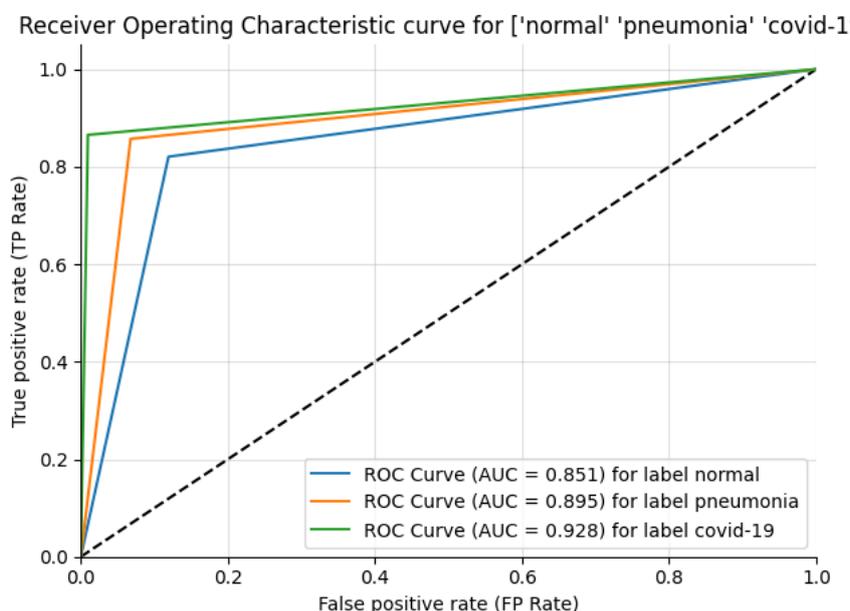


Figura 32: Curva ROC del modelo COVID-X-NET

Métrica de evaluación del modelo	Resultado
Error estimado del método de clasificación	87
Error estimado relativo del método de clasificación	0.1462
Precisión multi clase	0.8538
Precisión por clase	N: 0.6274, P:0.8615, Y:0.9878
F1 score por clase	N:0.7111, P:0.8593, Y:0.9224
Tasa de verdaderos positivos por clase	N:0.8205, P: 0.8571, Y:0.8652
Tasa de falsos positivos por clase	N:0.1192, P:0.0676, Y:0.0095
Sensibilidad	N:0.8205, P: 0.8571, Y:0.8652
Exactitud multi clase	0.8537
Exactitud por clase	N:0.8689, P:0.9076, Y:0.9311

Tabla 5: Resultados de las métricas de evaluación para el modelo COVID-X-NET.

### 6.1.0.2 Evaluación de COVID-X-CLINICAL

La evaluación de este modelo, basado en *Random Forest*, se detalla en la Tabla 6. Estos resultados corresponden con la aplicación de la metodología descrita anteriormente utilizando 30% del conjunto de datos para pruebas.

En una primera instancia, se determinan los hiper parámetros mas adecuados para la configuración del modelo. Para ello la metodología de búsqueda

en grilla es utilizada. Esta consiste en hacer un entrenamiento con cada posible combinación de hiper parámetros.

Para el entrenamiento y evaluación de estos, se emplea el método *K-fold*, en el que se divide el conjunto de entrenamiento en  $K$  conjuntos. El entrenamiento se efectúa en  $K - 1$  conjuntos, y luego, la validación, en el restante. Este método permite que la evaluación de los modelos no este sesgada a los datos utilizados para ella, por ello se conoce el método *K-fold* como un método de validación cruzada.

De las métricas presentadas en suma con la representación visual de la matriz de confusión en la Figura 33, es posible notar que este modelo puede diferenciar de manera más sencilla los casos donde no se presentan síntomas de neumonía. Dada la curva *ROC*, se construye para cada clase en contraste con las demás en la Figura 34. En ella se nota una baja sensibilidad o tasa de verdaderos positivos para los casos clasificados de COVID-19. Sin embargo, la clasificación entre neumonía o no presenta hallazgos, no tiene tan baja sensibilidad. Es decir, aquí se ve que los casos positivos son muy difíciles de conseguir por esta información, pero es relativamente sencillo detectar casos negativos.

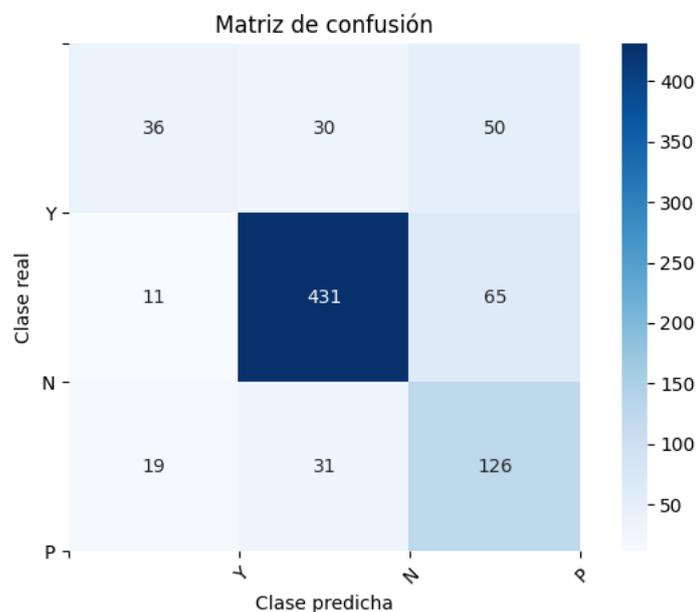


Figura 33: Matriz de confusión del modelo COVID-X-CLINICAL

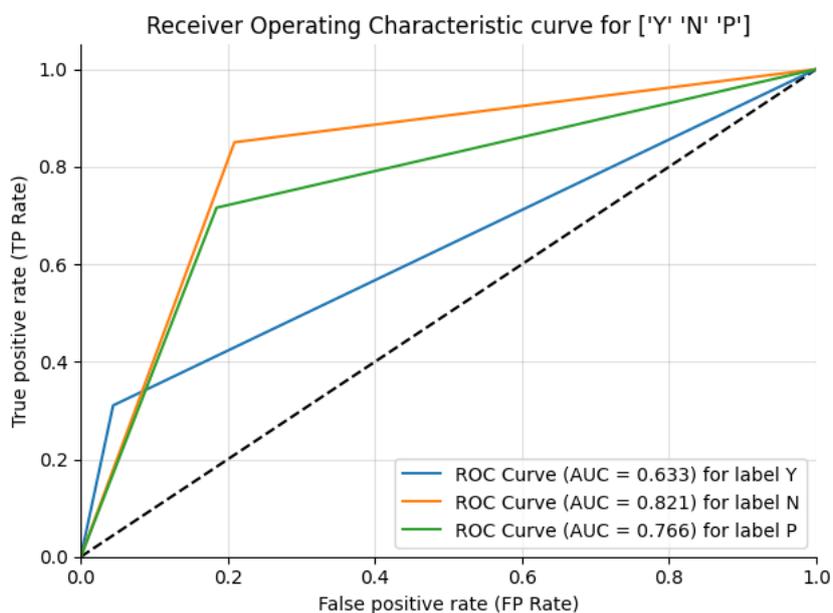


Figura 34: Curva ROC del modelo COVID-X-CLINICAL

Métrica de evaluación del modelo	Resultado
Error estimado del método de clasificación	206
Error estimado relativo del método de clasificación	0.2578
Precisión multi clase	0.7422
Precisión por clase	N:0.8760, P:0.5228, Y:0.5454
F1 score por clase	N:0.8629, P: 0.6043, Y:0.3956
Tasa de verdaderos positivos por clase	N:0.8501, P:0.7159, Y:0.3103
Tasa de falsos positivos por clase	N:0.2089, P:0.1846, Y:0.0439
Sensibilidad	N:0.8501, P:0.7159, Y:0.3103
Exactitud multi clase	0.7422
Exactitud por clase	N:0.8285, P:0.7935, Y:0.8623

Tabla 6: Resultados de las métricas de evaluación para el modelo COVID-X-CLINICAL.

### 6.1.0.3 Evaluación de COVID-X-AI

La evaluación de este modelo compuesto del modelo conjunto de redes neuronales artificiales convolucionales y el modelo basado en *Random Forest*, se detalla en la Tabla 7.

El Modelo de COVID-X-AI se entrenó utilizando los datos provistos por la universidad de Waterloo conformando parte del conjunto mencionado en

la sección de entrenamiento que se llama *Dataset image and clinical data*, del *dataset* original se destina un 70% para confeccionar el conjunto de entrenamiento para aprender la conjunción de estos dos modelos. El porcentaje restante conforma el conjunto de pruebas utilizado para la evaluación del modelo expuesta en esta sección.

Cabe destacar que el único conjunto de datos disponible que tiene, tanto las radiografías, como los datos clínicos asociados a la misma, es este conjunto, publicado por los investigadores de *Waterloo*. Es muy escaso, y por este motivo no se pone en producción el modelo conjunto, dado que como se observa en la curva ROC de la Figura 36, el clasificador resultante es menos preciso para nuestro caso de uso que si se utiliza el modelo COVID-X-NET.

El área debajo de la curva ROC mencionada, es considerablemente menor que en el modelo COVID-X-NET. Esto se lo puede ver como un *underfitting* del modelo sobre el *dataset* que se encuentra disponible, y por lo tanto, es necesario ampliar este para obtener un mejor resultado. Consecuentemente, al día de hoy se utiliza el modelo COVID-X-NET, y se esta en búsqueda de nuevas integraciones para poder seguir creciendo el conjunto de datos recolectados por el sistema COVID-X, y así, permitir realizar futuros entrenamientos, evaluación y análisis sobre nuevos modelos compuestos, u otros modelos de inteligencia artificial.

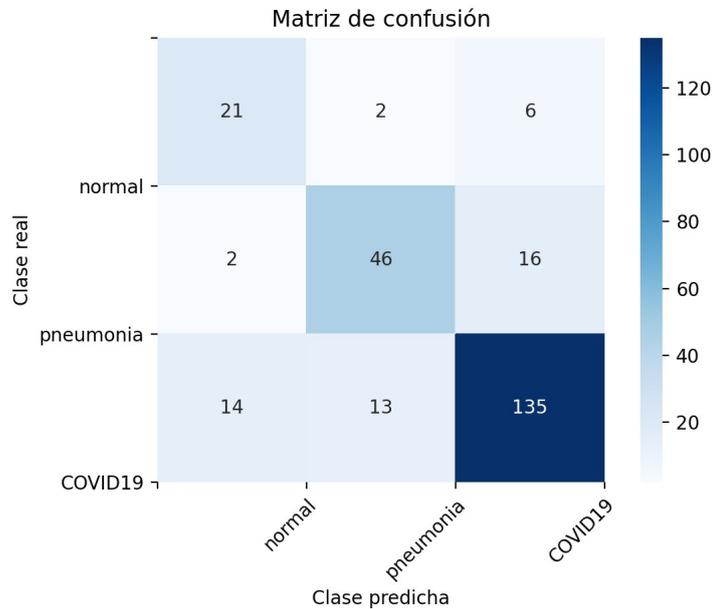


Figura 35: Matriz de confusión del modelo COVID-X-AI

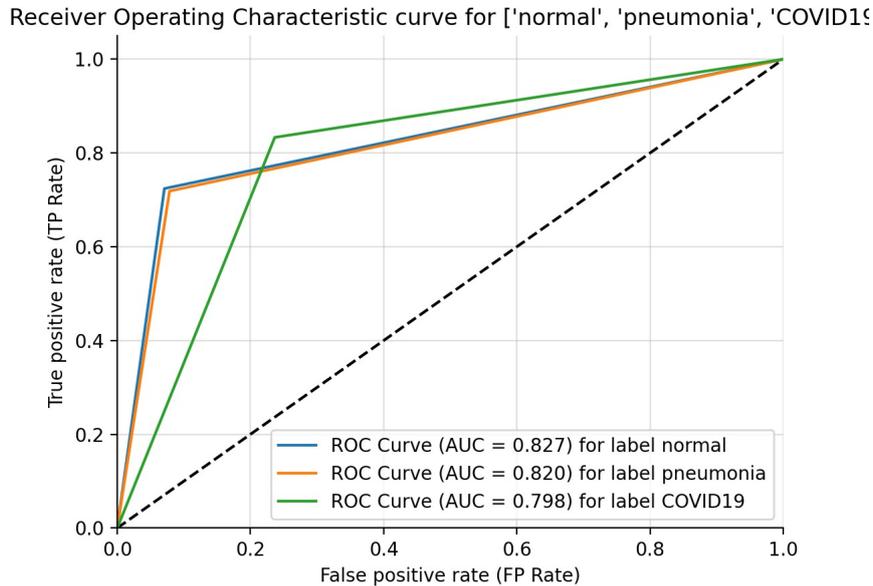


Figura 36: Curva ROC del modelo COVID-X-AI

Métrica de evaluación del modelo	Resultado
Error estimado del método de clasificación	53
Error estimado relativo del método de clasificación	0.2078
Precisión multi clase	0.7922
Precisión por clase	N:0.5676, P:0.7541, Y:0.8599
F1 score por clase	N:0.6364, P:0.7360, Y:0.8464
Tasa de verdaderos positivos por clase	N:0.7241, P:0.7187, Y:0.8333
Tasa de falsos positivos por clase	N:0.0707, P:0.0785, Y:0.2365
Sensibilidad	N:0.7241, P:0.7187, Y:0.8333
Exactitud multi clase	0.7922
Exactitud por clase	N:0.9059, P:0.8706, Y:0.8078

Tabla 7: Resultados de las métricas de evaluación para el modelo COVID-X-AI.

## 6.2 Integraciones

Al quedar disponible el producto desarrollado, el sistema COVID-X despertó interés en profesionales de la salud que decidieron formar parte de COVID-X. Dentro de los usuarios del sistema se destacan dos variantes: aquellos que pertenecen a una organización, y al conocer el sistema deci-

den formar parte del mismo por su cuenta, y por otra parte, los usuarios resultantes de un convenio o integración formal con alguna institución.

El lo que respecta a este primer grupo de usuarios independientes, que han mostrado interés por conocer y usar la plataforma COVID-X, se han podido registrar varios de ellos exitosamente en el sistema. Algunas de las organizaciones a las que pertenecen estas personas, que por propia iniciativa se registraron, son: Hospital Justo José de Urquiza, Clínica Humana de Imágenes, Centro de Imágenes Médicas 25 de Mayo, Hospital JM Cullen, entre otras.

Por otra parte, en lo que refiere a las integraciones formales, durante las primeras versiones del sistema se comenzó un vínculo con la clínica de imágenes Leben Salud. En este contexto, se concretó un convenio marco entre el ITBA y Leben, y luego hemos contado con la colaboración de Leben brindando el apoyo de sus profesionales para llevar a cabo varias tareas esenciales en el desarrollo del sistema. Finalmente, la empresa *Salesforce*, organización a la que pertenecen Santiago Bassani y Juan Li Puma, uno de los colaboradores principales del proyecto, tomo un rol de mediador para la apertura del dialogo con la institución *UCSF Benioff Children's Hospital*.

## 7 Manual de uso

En esta sección se busca describir cómo son las funcionalidades que ofrece la aplicación web de COVID-X. Al ingresar al sitio <https://www.cov-x.com> se puede observar la siguiente pantalla de inicio:

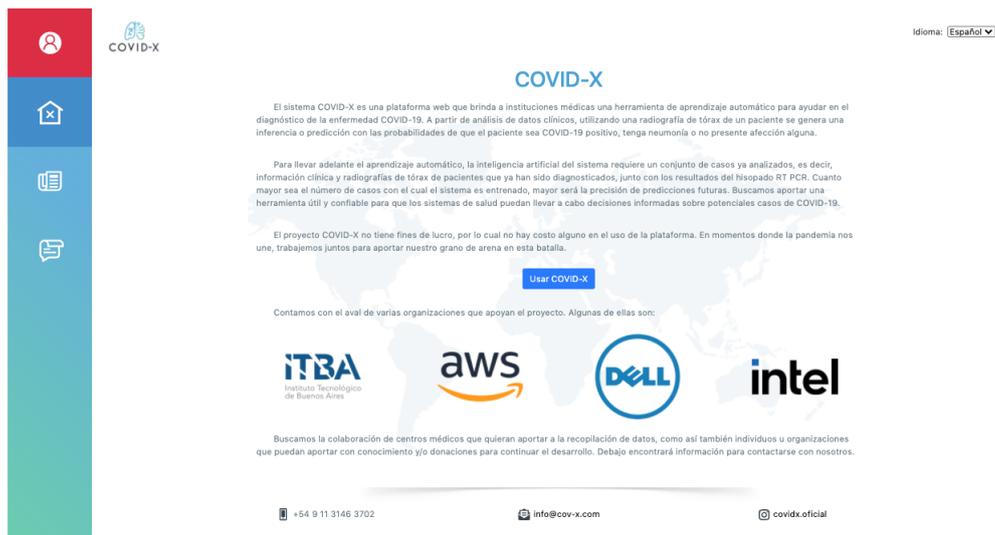


Figura 37: Pantalla principal de la aplicación web COVID-X.

### 7.1 Ingreso y registro de nuevo usuario

Al querer utilizar el sistema COVID-X, la aplicación se dirige a la pantalla de ingreso y registro de nuevo usuario como se observa en la Figura 38. Los nuevos usuarios pasan un riguroso proceso de verificación de datos para garantizar que estos sean personas de las organizaciones de la salud que se integran con COVID-X. Por ello el proceso de registro de un nuevo usuario puede demorar un par de días, hasta que el mismo es dado de alta en el sistema. Al momento de registrarse, también se indica si el usuario solo utilizará COVID-X para hacer carga de datos de entrenamiento, para realizar inferencias o ambas, definiendo así su rol de usuario dentro del sistema.

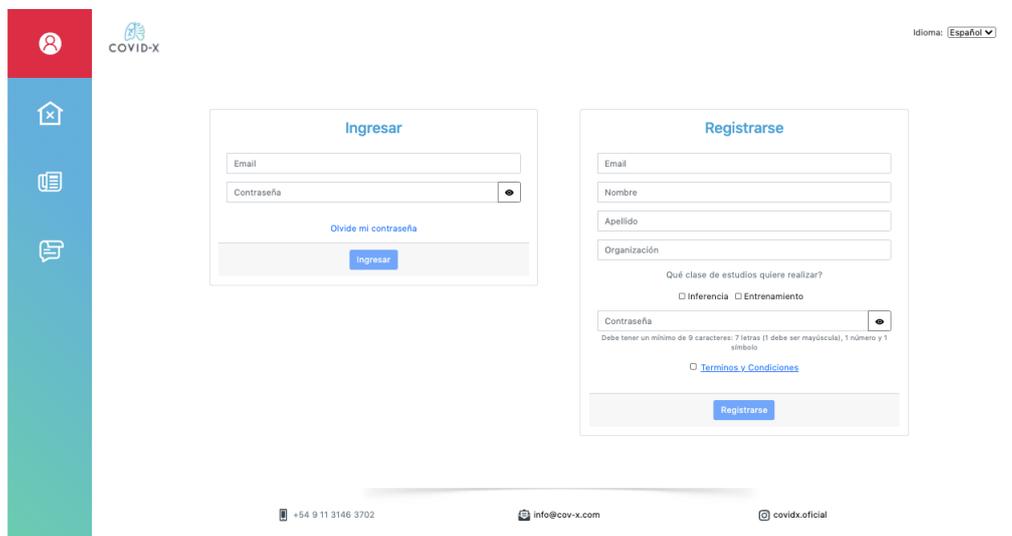


Figura 38: Pantalla de ingreso y registro de nuevo usuario.

Al registrar un nuevo usuario, este recibe un correo electrónico para verificar la información de contacto del usuario, similar al siguiente:



Figura 39: Correo electrónico de verificación de datos de contacto.

Luego de verificar el correo electrónico exitosamente, el sistema envía al usuario un segundo correo confirmando la verificación e informando que la activación del usuario queda pendiente hasta que un moderador valide los datos del usuario. En la Figura 40, se observa un correo como el mencionado.



Figura 40: Correo electrónico de verificación de datos de contacto exitosa.

Finalmente, una vez que un administrador de COVID-X valide los datos y active el usuario desde el backoffice de moderación, se envía un tercer correo electrónico al usuario notificando que su cuenta está lista para utilizar el sistema. La Figura 41 ilustra este tercer correo.



Figura 41: Correo electrónico de activación exitosa.

Una vez que se posee un usuario aprobado para utilizar el sistema, se puede realizar el ingreso, que de ser este exitoso se observará un mensaje que lo informa como se puede ver en la Figura 42.

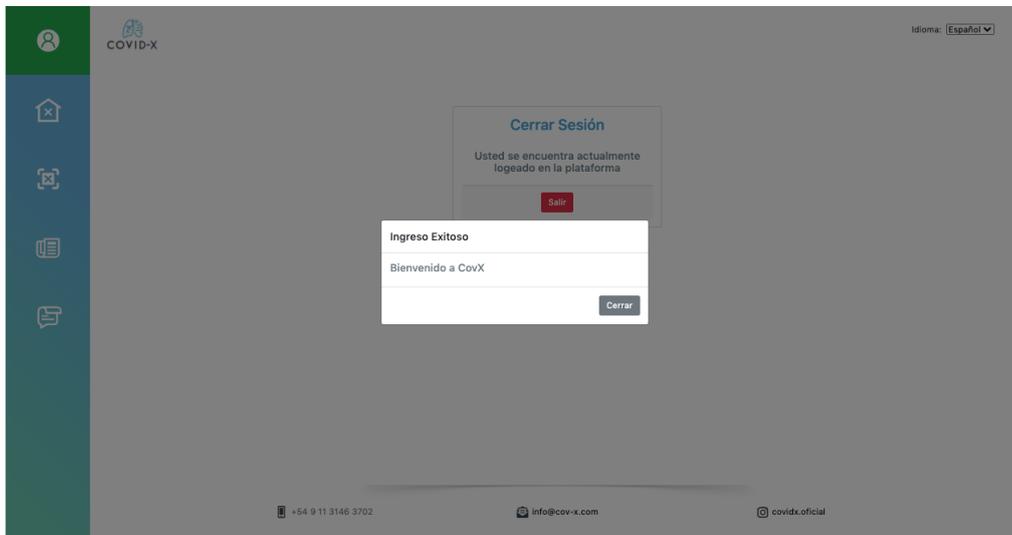


Figura 42: Mensaje de ingreso exitoso.

Ahora bien, en función de lo indicado en el proceso de registro, el usuario que ingresa en el sistema puede ver la o las opciones de Inferencia y/o entrenamiento como se muestra en la Figura 43.

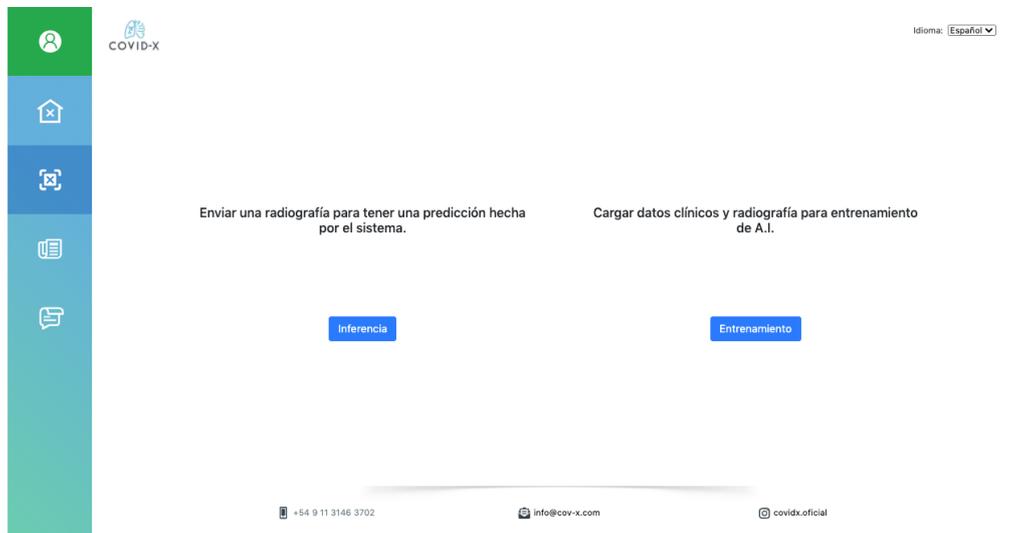


Figura 43: Pantalla con los flujos principales de COVID-X.

## 7.2 Carga de datos para entrenamiento

En caso de utilizar la función de carga de datos para entrenamiento, se presenta un formulario como se observa en la Figura 44, para la carga de

datos clínicos disponibles para el caso a registrar.

The image shows a web application interface for entering training data. On the left is a vertical navigation bar with icons for home, search, list, and chat. The main content area is titled "Entrenamiento" and includes a consent notice: "Les pedimos que descarguen este convenio para ser firmado por el paciente a quién pertenecerán los datos." Below this, a prompt says "Complete la información de la cual dispone." The form fields are: "Fecha de Nacimiento:" (date picker), "Sexo:" (radio buttons for Masculino, Femenino, Otro), "Temperatura:" (text input with °C unit), "Altura:" (text input with m unit), "Peso:" (text input with Kg unit), "¿Padece Asma?:" (dropdown), "¿Padece EPOC?:" (dropdown), "Resultado prueba RT-PCR:" (dropdown), "Presión parcial de Oxígeno:" (text input with mmHg unit), "Saturación O2:" (text input with % unit), "Frecuencia cardíaca:" (text input with PPM unit), "Frecuencia respiratoria:" (text input with RPM unit), "Presión arterial:" (text input with mmHg unit), and "Rayos X:" (file upload button labeled "Seleccionar archivo" with a note "No se editó archivo"). The footer contains contact information: "+54 9 11 3140 3702", "info@cov-x.com", and "covida.official".

Figura 44: Formulario para carga de datos de entrenamiento.

En caso de tratarse de un usuario de Argentina, se puede acceder a un formulario de consentimiento informado, como se observa en la Figura 45, para presentar y ser firmado por los pacientes que están de acuerdo con aportar sus datos para la constitución de los datos de entrenamiento.

## CONSENTIMIENTO INFORMADO

Fecha: \_\_ de \_\_\_\_\_ de \_\_\_\_\_

Lugar: \_\_\_\_\_

Yo \_\_\_\_\_ con Documento Nacional de Identidad Nro \_\_\_\_\_ en pleno uso de razón declaro haber sido informado por los funcionarios de la Institución \_\_\_\_\_ de la investigación que se está llevando a cabo con fines científicos, académicos u otras afines en el marco de la pandemia de covid-19. Para la cual se requiere la utilización de análisis y resultados de pacientes que hayan o no padecido la enfermedad covid-19, y que serán procesados por medio del sistema **COVID-X**. Asimismo, he realizado todas las preguntas que consideré pertinentes, habiendo recibido como respuesta explicaciones claras y detalladas con las que he quedado conforme.

Por ello, habiendo tomado conocimiento de tal investigación y con los fines de colaborar con la misma, por medio de la presente otorgo mi consentimiento libre y expreso para que se utilicen mis datos (radiografía de tórax y otros datos clínicos) con los fines expresados en el párrafo anterior y se procesen los mismos con el sistema **COVID-X**.

Firma: \_\_\_\_\_

Aclaración  
\_\_\_\_\_

D.N.I.: \_\_\_\_\_

Figura 45: Documento de consentimiento informado del uso de datos de pacientes en COVID-X para Argentina.

El Formulario para realizar la carga de datos de un nuevo estudio con propósito de entrenamiento incluye los siguientes campos:

1. Fecha de nacimiento
2. Sexo (Masculino / Femenino / Otro)
3. Temperatura corporal (en grados centígrados °C)
4. Altura (en metros)
5. Peso (en kilogramos)
6. ¿Tiene asma? (SI / NO / NO SABE)

7. ¿Tiene EPOC? (SI / NO / NO SABE)
8. Resultado prueba RT-PCR (POSITIVO / NEGATIVO)
9. Presión parcial de Oxígeno (en mmHg)
10. Saturación de Oxígeno (en %)
11. Frecuencia Cardíaca (en Latidos x minuto)
12. Frecuencia Respiratoria (en Respiraciones x minuto)
13. Presión Arterial (en mmHg)
14. Archivo DICOM con radiografía de Tórax (archivo con extensión .dcm)

Luego de completar el formulario, aparece una pantalla de espera mientras se procesan los datos como se puede ver en la Figura 47, y cuando finaliza la carga, un mensaje de confirmación de la carga de datos como se observa en la Figura 48.

Figura 46: Formulario para carga de datos de entrenamiento con datos de ejemplo.

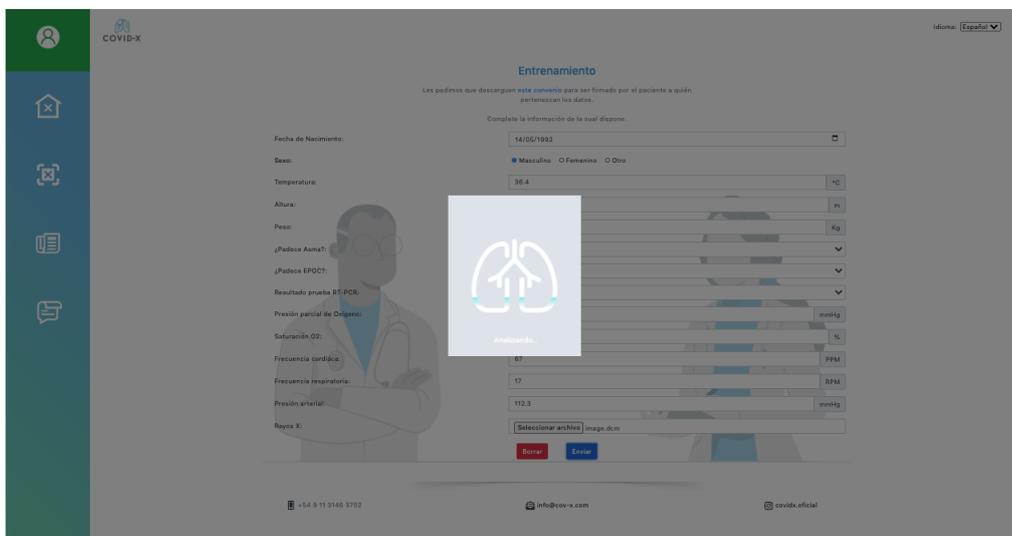


Figura 47: Pantalla durante el procesamiento de la carga de datos de entrenamiento.

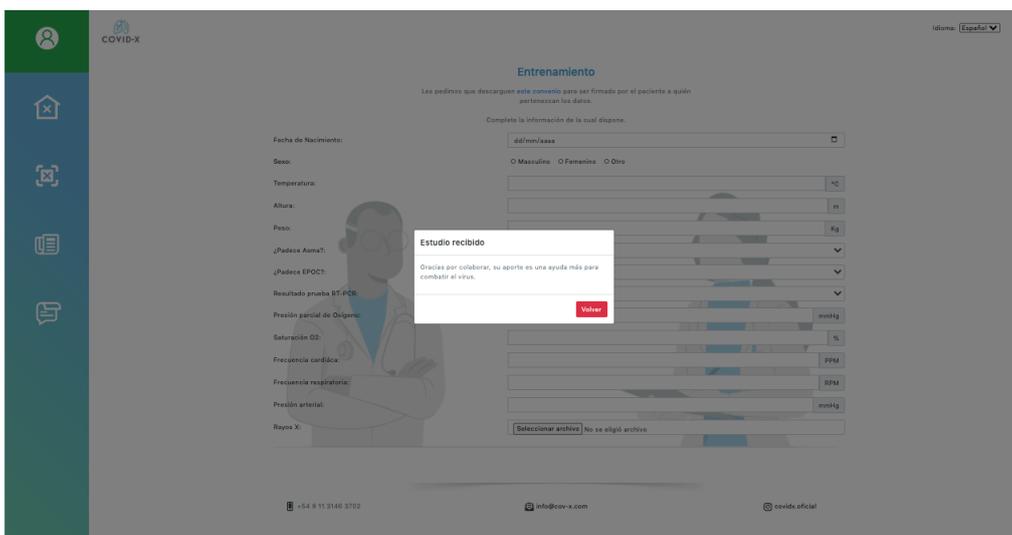


Figura 48: Pantalla de confirmación de estudio recibido en la carga de datos para entrenamiento.

### 7.3 Realizar una Inferencia

En caso de utilizar la función inferencia, se presenta un formulario como se observa en la Figura 49, para la carga de datos clínicos disponibles del caso a analizar para generar una predicción o inferencia.

Figura 49: Formulario para realizar una inferencia.

Nuevamente, en caso de tratarse de un usuario de Argentina, se puede acceder a un formulario de consentimiento informado, como se observa en la Figura 45, para presentar y ser firmado por los pacientes que están de acuerdo con aportar sus datos para realizar una inferencia.

El Formulario para realizar la carga de datos de un nuevo estudio con propósito de inferencia incluye los siguientes campos:

1. Fecha de nacimiento
2. Sexo (Masculino / Femenino / Otro)
3. Temperatura corporal (en grados centígrados °C)
4. Altura (en metros)
5. Peso (en kilogramos)
6. ¿Tiene asma? (SI / NO / NO SABE)
7. ¿Tiene EPOC? (SI / NO / NO SABE)
8. Presión parcial de Oxígeno (en mmHg)
9. Saturación de Oxígeno (en %)
10. Frecuencia Cardíaca (en Latidos x minuto)
11. Frecuencia Respiratoria (en Respiraciones x minuto)

12. Presión Arterial (en mmHg)

13. Archivo DICOM con radiografía de Tórax (archivo con extensión .dcm)

The screenshot shows the 'Inferencia' form in the COVID-X application. The form is titled 'Inferencia' and includes a disclaimer: 'Les pedimos que descarguen este convenio para ser firmado por el paciente a quién pertenecen los datos.' Below this, it states 'Todo dato extra incrementa la precisión del análisis'. The form fields are filled with the following example data:

Field	Value
Fecha de Nacimiento:	14/05/1992
Sexo:	Masculino
Temperatura:	36.4
Altura:	1.8
Peso:	75.5
¿Padeca Asma?:	Si
¿Padeca EPOCT?:	No
Presión parcial de Oxígeno:	84
Saturación O2:	98.8
Frecuencia cardíaca:	67
Frecuencia respiratoria:	17
Presión arterial:	112.3
Rayos X:	Seleccionar archivo   image.dcm

At the bottom of the form, there are 'Borrar' and 'Enviar' buttons. The footer of the application shows the phone number '+54 9 11 3146 3702', the email 'info@cov-x.com', and the social media handle '@covidaoficial'.

Figura 50: Formulario para realizar una inferencia con datos de ejemplo.

Luego de completar el formulario, aparece una pantalla de espera mientras se procesan los datos y se realiza la predicción, como se aprecia en la Figura 51.

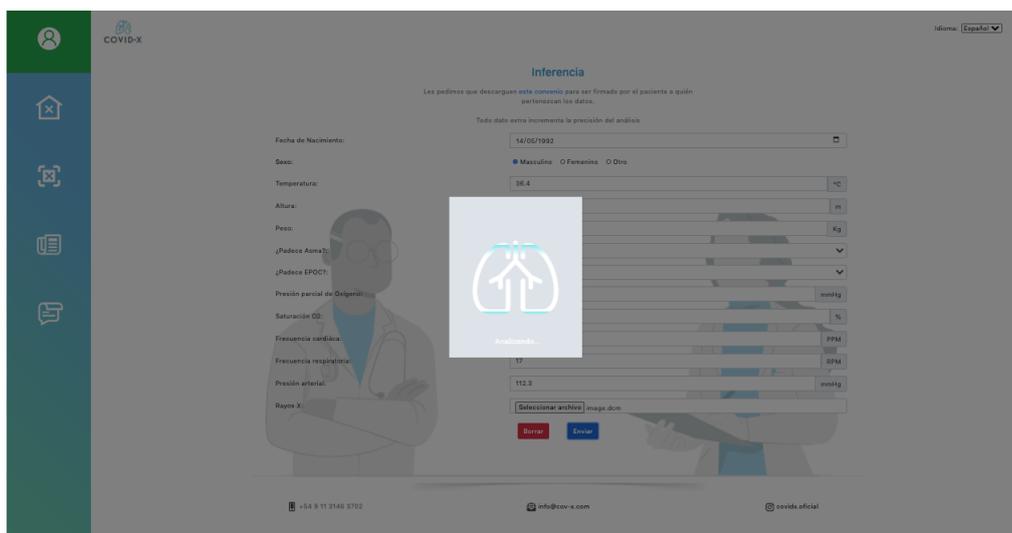


Figura 51: Pantalla durante el procesamiento y realización de una predicción en el proceso de inferencia.

Finalmente, al procesarse los datos y luego realizarse la predicción solicitada, los resultados de la inferencia se ilustran en un gráfico de torta donde muestran los porcentajes de las probabilidades de que el caso analizado corresponde con COVID-19, neumonía o si se trata de un caso normal sin ninguna de estas afecciones, como se puede ver en la Figura 52.

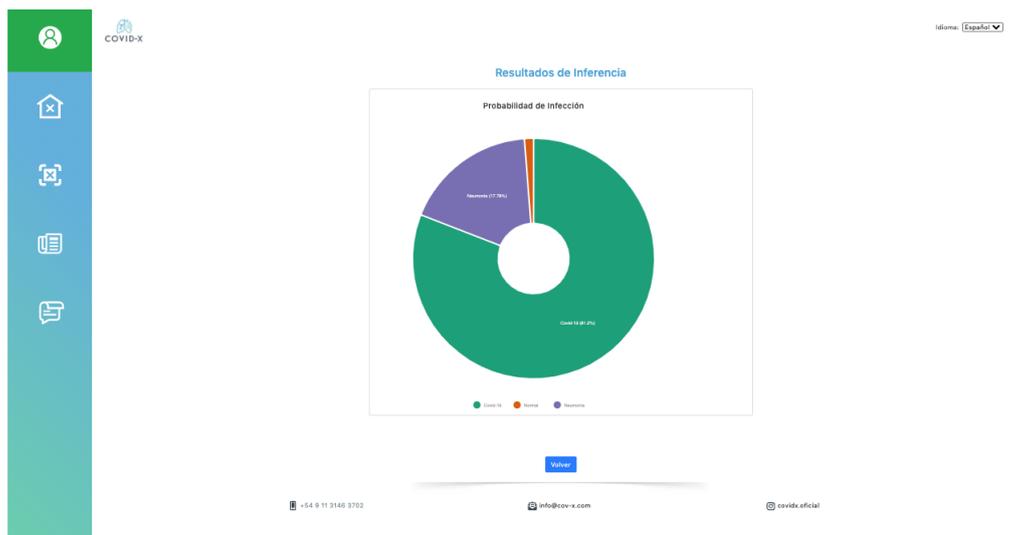


Figura 52: Pantalla de resultados de la inferencia.

## 7.4 Fundamentos para la investigación

En esta sección del sitio web se busca poder exponer artículos de fuentes externas relacionados con el proyecto COVID-X, así como también de su investigación y desarrollo. Esta sección también incluye enlaces de utilidad en el contexto de la pandemia del COVID-19 y los términos y condiciones de uso del sistema. En la Figura 53 se observa la pantalla del sitio correspondiente a esta sección.

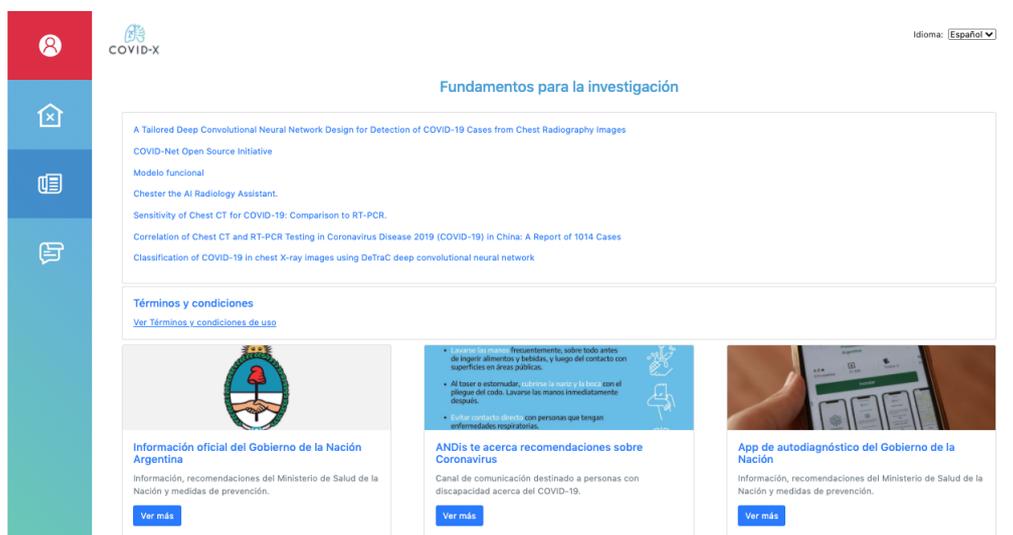


Figura 53: Pantalla fundamentos para la investigación.

## 7.5 COVID-X Blog

La aplicación cuenta con una sección que conforma un blog llamado COVID-X Blog. El objetivo de este blog es agrupar un conjunto de entradas que sirvan como canal para comunicación del desarrollo del proyecto, novedades y decisiones que se toman con respecto a la plataforma. En la Figura 54 se puede observar la vista de la pantalla del blog.

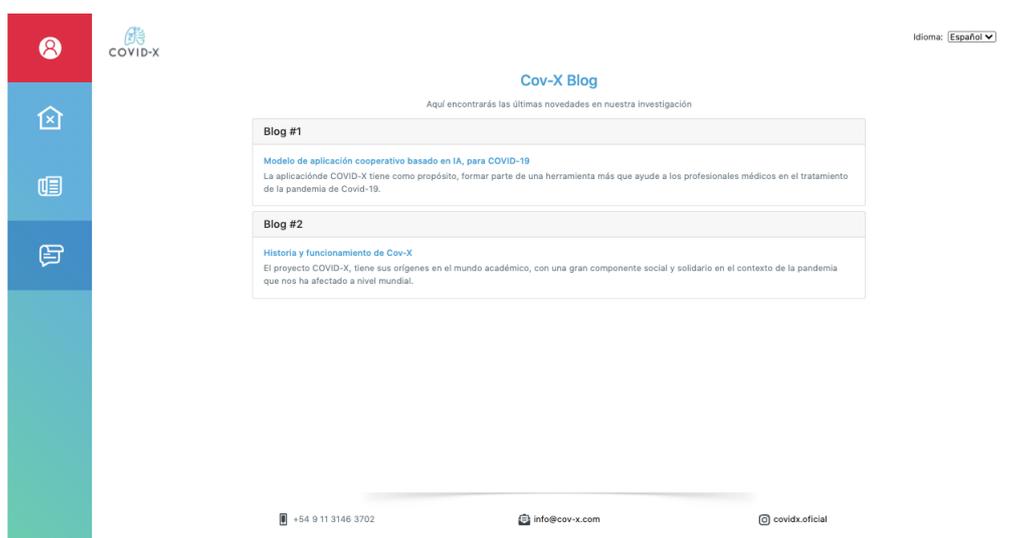


Figura 54: Pantalla de COVID-X Blog.

## 7.6 Recuperar contraseña

Como se puede observar en la Figura 38, al momento de querer ingresar al sistema, se dispone de un botón "Olvide mi contraseña" que comienza el proceso de recupero de contraseña para una cuenta activada de COVID-X. En la Figura 55 se observa la pantalla que actúa como punto de entrada para comenzar el proceso de recuperación.

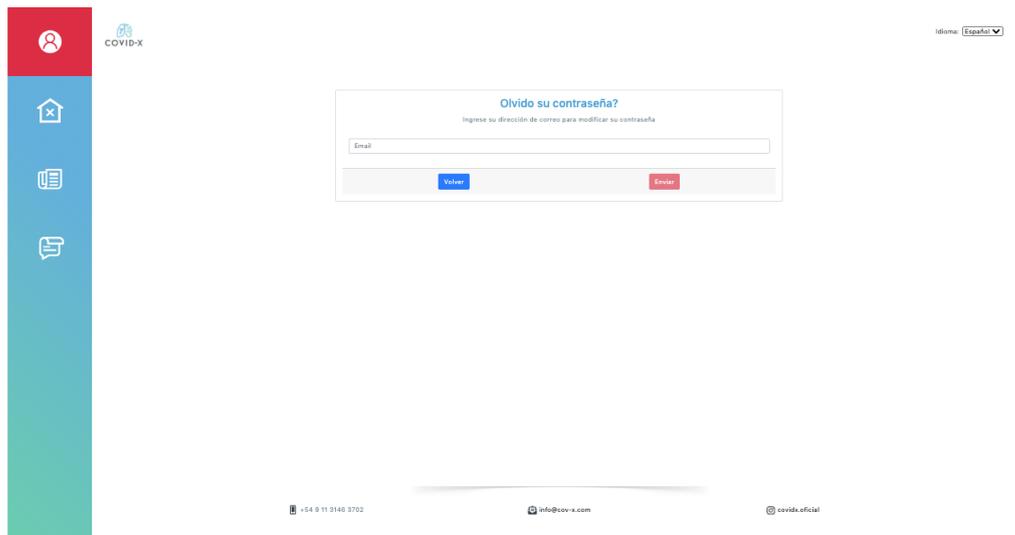


Figura 55: Pantalla de recupero de contraseña.

Como se mencionó anteriormente, es requerido que el usuario que desea comenzar el recupero de su contraseña disponga de una cuenta de usuario con datos de contacto previamente verificados y además que su cuenta haya sido activada. Si el usuario cumple estos requisitos, puede proceder a ingresar su correo electrónico en el formulario y continuar. Confirmando el comienzo de este proceso se observa una pantalla como la de la Figura 56

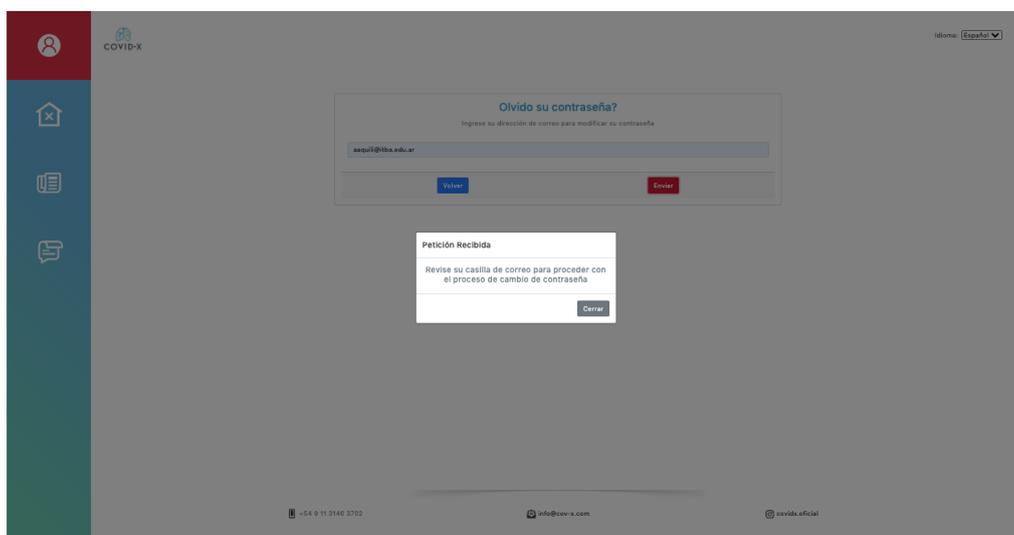


Figura 56: Mensaje de petición de cambio de contraseña recibida.

Luego de procesar la petición enviada, el sistema envía al usuario un correo electrónico para crear una nueva contraseña. Un ejemplo de correo electrónico del sistema se puede visualizar en la Figura 57.



Figura 57: Correo electrónico para cambiar contraseña.

Al presionar en el botón de "Crear nueva contraseña", se ejecuta una redirección hacia una pantalla con un formulario para ingresar la nueva contraseña. La pantalla mencionada se ilustra en la Figura 58.

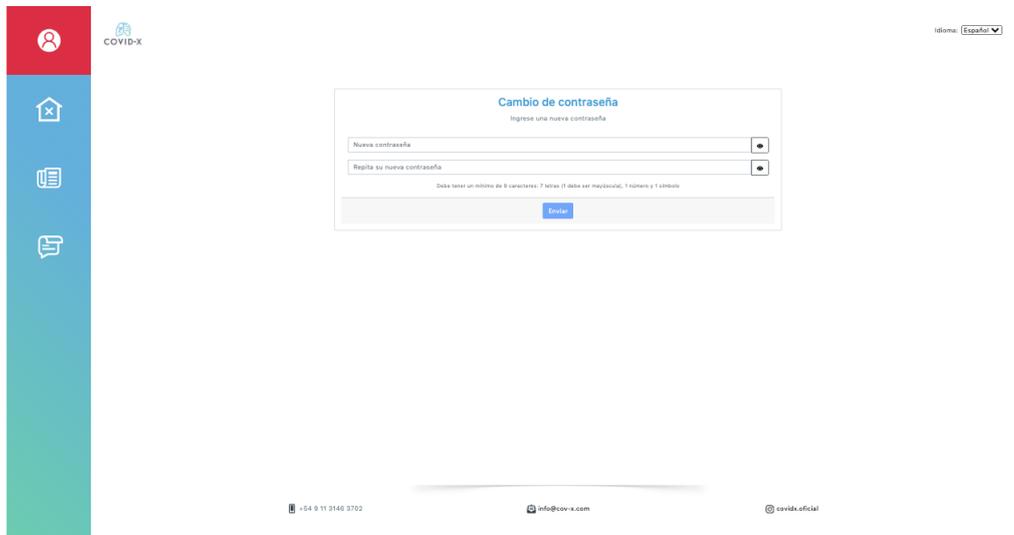


Figura 58: Pantalla de cambio de contraseña.

Se debe ingresar dos veces una nueva contraseña que cumpla con las mismas restricciones impuestas al momento de registrar un nuevo usuario. Luego, al enviar el formulario de cambio de contraseña, se visualiza en pantalla un mensaje como el de la Figura 59.

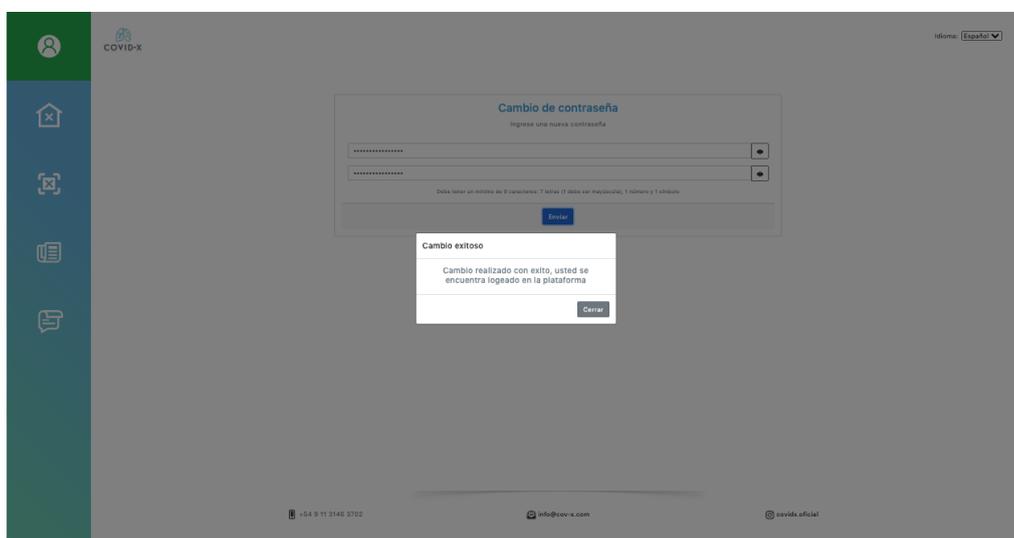


Figura 59: Mensaje de cambio de contraseña exitoso.

## 7.7 Salir del sistema

Finalmente, para salir del sistema, en la sección de usuario se encuentra el botón de cerrar sesión. En la Figura 60 se observa la pantalla de la sección de usuario con el botón para salir del sistema.

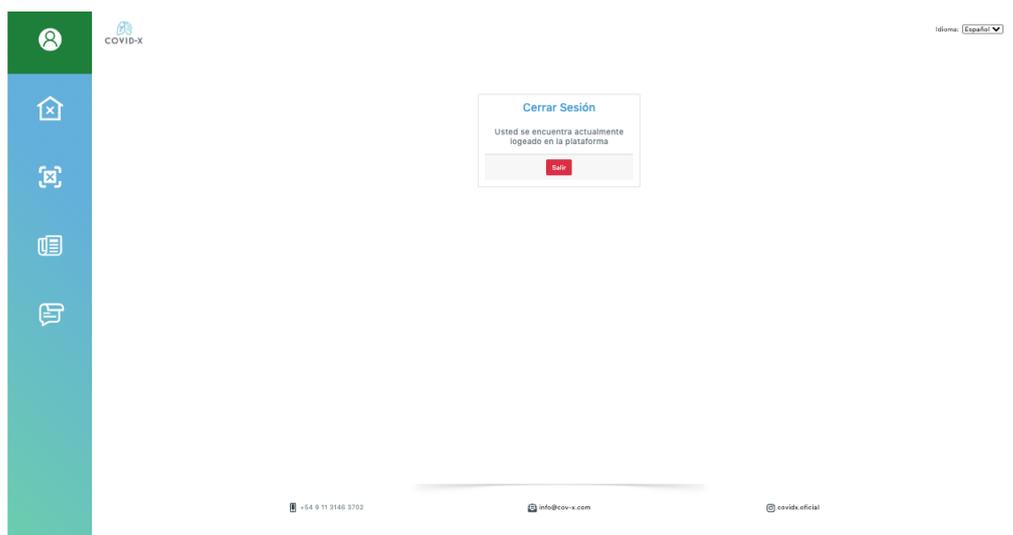


Figura 60: Pantalla para salir del sistema.

## 8 Conclusiones y trabajo futuro

En el presente trabajo se presentó un sistema informático desde su diseño hasta su puesta en producción, pasando por todas sus etapas intermedias. La idea de llevar a las personas una solución a un problema del mundo real, teniendo el potencial de gran utilidad en el contexto actual de la pandemia del COVID-19, genera una gran motivación a los autores en búsqueda de realizar un aporte en la crisis sanitaria, no solo para personas de Argentina, sino para toda persona en el mundo.

En una gran medida, las habilidades adquiridas en la formación académica en la carrera de grado de ingeniería en informática se han puesto en uso para llevar este proyecto a cabo, incluyendo los aspectos de diseño, desarrollo e implementación, planificación y gestión de proyectos informáticos. También requiriendo la aplicación de habilidades de carácter más técnico para la implementación de los subsistemas y la administración de la infraestructura global.

De ninguna manera es una solución planteada para ser un método de prueba en el diagnóstico de la enfermedad COVID-19. La esperanza es que los resultados prometedores logrados por el sistema de IA, con el conjunto de datos de prueba, abre las puertas a que los resultados sean mejores y más confiables conforme se logre expandir este conjunto. Haciendo uso del mismo sistema que permite la recolección de datos, a futuro, se puede seguir creciendo en el número de ejemplares de estudios y datos clínicos almacenados bajo el sistema lo cual podría ser utilizado para posteriores investigaciones y aplicaciones. El campo de acción de estos datos no se limita solo a la enfermedad COVID-19, con lo cual el crecimiento de esta base de datos es independiente de la enfermedad mencionada.

Se espera que con el transcurso del tiempo, se puedan efectuar nuevas integraciones de instituciones u organizaciones médicas con el sistema COVID-X, en búsqueda de aportar valor a los usuarios del sistema y al mismo tiempo tener la posibilidad de obtener más radiografías de tórax junto con sus correspondientes datos clínicos asociados, y de esta manera, aumentar el tamaño del conjunto de datos.

A su vez, al mismo tiempo que se busquen concretar nuevas integraciones, es posible continuar iterando la solución propuesta, obteniendo mayor retroalimentación por parte de los nuevos usuarios especializados. También es posible ampliar el sistema al uso de otros tipos de imágenes y otro tipo de análisis médico de las mismas.

## Referencias bibliográficas

- [1] K. Beck, *Extreme programming explained: Embrace change*, The XP series, Addison-Wesley, 1999.
- [2] Andre Esteva *et al.*, *Dermatologist-level classification of skin cancer with deep neural networks*, Nature **542** (2017), no. 7639, 115–118.
- [3] Clara Bonanad *et al.*, *The effect of age on mortality in patients with covid-19*, Journal of the American Medical Directors Association **21** (2020), no. 7, 915 – 918.
- [4] Huang P *et al.*, *Added value of computer-aided ct image features for early lung cancer diagnosis with small pulmonary nodules: A matched case-control study*, Radiology **1** (2018), no. 286, 286–295.
- [5] Md Martuza Ahamad *et al.*, *A machine learning model to identify early stage symptoms of sars-cov-2 infected patients*, Expert systems with applications **160** (2020), 113661–113661.
- [6] Michael Chung *et al.*, *Ct imaging features of 2019 novel coronavirus (2019-ncov)*, Radiology **295** (2020), no. 1, 202–207.
- [7] Pranav Rajpurkar *et al.*, *CheXnet: Radiologist-level pneumonia detection on chest x-rays with deep learning*, CoRR **abs/1711.05225** (2017).
- [8] Tao Ai *et al.*, *Correlation of chest ct and rt-pcr testing for coronavirus disease 2019 (covid-19) in china: A report of 1014 cases*, Radiology **296** (2020), no. 2, E32–E40.
- [9] Tian Sijia *et al.*, *Characteristics of covid-19 infection in beijing*, Journal of Infection **80** (2020), no. 4, 401 – 406.
- [10] Varun Gulshan *et al.*, *Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs*, JAMA **316** (2016), no. 22, 2402–2410.
- [11] E. Evans and M. Fowler, *Domain-driven design: Tackling complexity in the heart of software*, Addison-Wesley, 2004.
- [12] Idevaldo *et al.* Floriano, *Accuracy of the polymerase chain reaction (pcr) test in the diagnosis of acute respiratory syndrome due to coronavirus: a systematic review and meta-analysis*, Rev. Assoc. Med. Bras **66** (2020), 880 – 888.

- [13] Seth Gilbert and Nancy Lynch, *Brewer’s conjecture and the feasibility of consistent, available, partition-tolerant web services*, **33** (2002), no. 2, 51–59.
- [14] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, *Mobilenets: Efficient convolutional neural networks for mobile vision applications*, 2017.
- [15] John Ioannidis, *The infection fatality rate of covid-19 inferred from seroprevalence data*, medRxiv (2020).
- [16] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silvana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng, *CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison*, 2019.
- [17] Mohammad Tariqul Islam, Md Abdul Aowal, Ahmed Tahseen Minhaz, and Khalid Ashraf, *Abnormality detection and localization in chest x-rays using deep convolutional neural networks*, CoRR **abs/1705.09850** (2017).
- [18] Adam Jacobi, Michael Chung, Adam Bernheim, and Corey Eber, *Portable chest x-ray in coronavirus disease-19 (covid-19): A pictorial review*, *Clinical imaging* **64** (2020), 35–42.
- [19] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, *What is the best multi-stage architecture for object recognition?*, 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 2146–2153.
- [20] Philippe Kruchten, *Reference: Title: Architectural blueprints—the “4+1” view model of software architecture*, *IEEE software* **12** (1995), 6.
- [21] Sundaram B Lakhani P, *Deep learning at chest radiography: Automated classification of pulmonary tuberculosis by using convolutional neural networks*, *Radiology* **2** (2017), no. 284, 574–582.
- [22] Yann Lecun, Koray Kavukcuoglu, and Clement Faret, *Convolutional networks and applications in vision*, 05 2010, pp. 253–256.

- [23] Ana Luisa Mendizábal Méndez, *Radiación ionizante en tomografía computada: un tema de reflexión*, Anales de Radiología México **2** (2012), 90–97.
- [24] Ming-Yen *et al.* Ng, *Imaging profile of the covid-19 infection: Radiologic findings and literature review*, Radiology: Cardiothoracic Imaging **2** (2020), no. 1, e200034.
- [25] Raquel Sánchez-Oro, Julio Torres Nuez, and Gloria Martínez-Sanz, *Radiological findings for diagnosis of sars-cov-2 pneumonia (covid-19)*, Medicina clinica **155** (2020), no. 1, 36–40 (eng), 32321656[pmid].
- [26] Linda Wang and Alexander Wong, *Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images*, 2020.

# Anexos

## Anexo A Documentación de APIs REST

En este anexo se presenta la documentación de los principales *endpoints* de los servicios que componen el *backend* de COVID-X.

### A.1 COVID-X-USERS

#### A.1.0.1 Registro de un nuevo usuario

**Ruta:** /covid-x/users

**Acceso:** Público

**Método HTTP:** POST

**Cuerpo de la petición HTTP:**

```
{
  "first_name": <FIRST-NAME>,
  "last_name": <LAST-NAME>,
  "email": <EMAIL>,
  "organization": <ORGANIZATION>,
  "password": <PASSWORD>,
  "roles": [<ROLE-1>, ... , <ROLE-N>]
}
```

- **<FIRST-NAME>**: Nombre del usuario(**String**)(**requerido**)  
(entre 1 y 100 caracteres)
- **<LAST-NAME>**: Apellido del usuario(**String**)(**requerido**)  
(entre 1 y 100 caracteres)
- **<EMAIL>**: Correo electrónico del usuario (**String**) (**requerido**)
- **<ORGANIZATION>**: Organización del usuario(**String**)(**requerido**)  
(entre 1 y 100 caracteres)
- **<PASSWORD>**: Contraseña del usuario (**String**) (**requerido**)  
(entre 10 y 100 caracteres)(Mínimo 7 letras + 1 número + 1 mayúscula + 1 carácter especial)

- **<ROLE-1>, ..., <ROLE-N>**: Nombre de los roles del usuario  
(**String**)(**requerido**)  
(Nombre de los posibles roles: "covid-x-training-set-writer", "covid-x-inference-set-writer")

**Solicitud de ejemplo:**

```
curl --location --request POST
  'https://api.cov.x.com/covid-x/users' \
--header 'Content-Type: application/json' \
--data-raw '{
  "first_name": "Alejo",
  "last_name": "Aquili",
  "email": "aaquili@itba.edu.ar",
  "organization": "itba",
  "password": "Alejo@CovidX2020",
  "roles": ["covid-x-training-set-writer",
    "covid-x-inference-set-writer"]
}'
```

**Respuesta de ejemplo:**

```
{
  "first_name": "Alejo",
  "last_name": "Aquili",
  "email": "aaquili@itba.edu.ar",
  "organization": "itba"
}
```

**Código de estado HTTP:** 201

### A.1.0.2 Ingreso de un usuario

**Ruta:** /covid-x/login

**Acceso:** Público

**Método HTTP:** POST

**Cuerpo de la petición HTTP:**



**Método HTTP:** GET

**Solicitud de ejemplo:**

```
curl --location --request GET
  'https://api.cov.x.com/covid-x/users' \
--header 'Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9'
```

**Respuesta de ejemplo:**

```
{
  "first_name": "Alejo",
  "last_name": "Aquila",
  "email": "aaquila@itba.edu.ar",
  "organization": "itba"
}
```

**Código de estado HTTP:** 200

**Respuesta erronea de ejemplo:**

```
{
  "error_code": "unauthorized",
  "description": "Missing_or_invalid_credentials",
  "status": 401
}
```

**Código de estado HTTP:** 401

#### A.1.0.4 Restablecer contraseña

**Ruta:** /covid-x/users/reset-passw

**Acceso:** Publico

**Método HTTP:** POST

**Cuerpo de la petición HTTP:**

```
{
  "email": <EMAIL>
}
```

- <EMAIL>: Correo electrónico del usuario que desea recuperar su contraseña (**String**) (**requerido**)

**Solicitud de ejemplo:**

```
curl --location --request POST
      'https://api.cov.x.com/covid-x/users/reset-password'
\
--header 'Content-Type: application/json' \
--data-raw '{
  "email": "aaquili@itba.edu.ar"
}'
```

**Respuesta de ejemplo:** Cuerpo vacío. **Código de estado HTTP:** 200

#### A.1.0.5 Cambio de contraseña

**Ruta:** /covid-x/users/change-password

**Acceso:** Público

**Método HTTP:** POST

**Parámetros:**

email=<EMAIL>

token=<TOKEN>

**Cuerpo de la petición HTTP:**

```
{
  "password": <NEW-PASSWORD>
}
```

- <EMAIL>: Correo electrónico del usuario (**String**)(**requerido**)  
(Proveniente del correo de recuperación enviado al usuario)

- **<TOKEN>**: Token de cambio de contraseña con un **UUID** aleatorio de 36 caracteres  
(**String**)(**requerido**)  
(Proveniente del correo de recuperación enviado al usuario)
- **<NEW-PASSWORD>**: Contraseña del usuario (**String**)(**requerido**)  
(entre 10 y 100 caracteres)(Mínimo 7 letras + 1 número + 1 mayúscula + 1 carácter especial)

**Solicitud de ejemplo:**

```
curl --location --request POST
  'https://api.covid-x.com/covid-x/users/change-password
?email=aaquili@itba.edu.ar
&token=9fc3d578-7616-431f-8df6-94ffb0b8d1b7' \
--header 'Content-Type: application/json' \
--data-raw '{
  "password": "Alejo@CovidX2021"
}'
```

**Respuesta de ejemplo:**

```
{
  "user_roles": ["covid-x-training-set-writer",
    "covid-x-inference-set-writer"],
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eo"
```

**Código de estado HTTP:** 200

**A.1.0.6 Activación de un usuario**

**Ruta:** /covid-x/users/userId/activate

**Acceso:** Privado, Rol requerido: "covid-x-admin"

**Método HTTP:** PUT

**Solicitud de ejemplo:**

```
curl --location --request PUT
  'https://api.cov-x.com/covid-x/users/1/activate' \
--header 'Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9' \
--header 'Accept-Language: es'
```

**Respuesta de ejemplo:** Cuerpo vacío. **Código de estado HTTP:** 204

## A.2 DICOMVERTER

### A.2.0.1 Procesar imagen DICOM

Este endpoint analiza y extrae metadatos del fichero DICOM, convierte la imagen a BMP y envía el fichero al PACS. **Ruta:** /dicomverter

**Acceso:** Privado

**Método HTTP:** POST

**Campos de formulario de petición HTTP:**

```
- dcm: <DCM-FILE>
```

- **<DCM-FILE>**: Binario del archivo DICOM (Archivo .dcm) (**requerido**)

**Solicitud de ejemplo:**

```
curl --location --request POST
  'https://dicomverter.cov-x.com/dicomverter' \
--header 'Content-Type: multipart/form-data' \
--form 'dcm=@/image.dcm'
```

Retorna en su cuerpo la imagen convertida en BMP y un encabezado HTTP *X-Study-Instance-Uid* con el *StudyInstanceUid* de la radiografía de tórax procesada. **Código de estado HTTP:** 200

## A.3 COVID-X-API

### A.3.0.1 Registrar un estudio para entrenamiento

Registra un estudio para el conjunto de datos de entrenamiento y retorna el *StudyInstanceUid* de la radiografía de tórax. **Ruta:** /covid-x/training

**Acceso:** Privado, Rol requerido: “covid-x-training-set-writer”

**Método HTTP:** POST

**Encabezados HTTP:** Authorization: Bearer <JWT-TOKEN>

**Campos de formulario de petición HTTP:**

```
- chest_x_ray_image: <DCM-FILE>
- birthday: <PATIENT-BIRTHDAY>,
- sex: <PATIENT-SEX>,
- temperature: <PATIENT-TEMPERATURE>,
- height: <PATIENT-HEIGHT>,
- weight: <PATIENT-WEIGHT>,
- has_asthma: <PATIENT-HAS-ASTHMA>,
- has_copd: <PATIENT-HAS-COPD>,
- oxygen_saturation: <PATIENT-OXYGEN-SAT>,
- test_rt_pcr: <TEST-RT-PCR>,
- partial_pressure_of_oxygen: <PATIENT-PO2>,
- heart_rate: <PATIENT-HEART-RATE>,
- breathing_frequency: <PATIENT-BREATHING-FREQ>,
- blood_pressure: <PATIENT-BLOOD-PRESSURE>
```

- <JWT-TOKEN>: Token JWT obtenido del login del usuario  
(String)(requerido)
- <DCM-FILE>: Binario del archivo DICOM (Archivo .dcm) (requerido)
- <PATIENT-BIRTHDAY>: Fecha de nacimiento del paciente  
(String)(RFC3339)(Opcional)
- <PATIENT-SEX>: El sexo del paciente, valores posibles [“MALE”,  
“FEMALE”, “OTHER”](String)(Opcional)
- <PATIENT-TEMPERATURE>: La temperatura del paciente al  
momento de realizarse el estudio en grados centígrados  
(Decimal)(Opcional)

- <**PATIENT-HEIGHT**>: La altura del paciente al momento de realizarse el estudio en metros (**Decimal**)(**Opcional**)
- <**PATIENT-WEIGHT**>: El peso del paciente al momento de realizarse el estudio en kilogramos (**Decimal**)(**Opcional**)
- <**PATIENT-HAS-ASTHMA**>: El paciente padece asma, valores posibles [“YES”, “NO”, “UNDEFINED”](**String**)(**Opcional**)
- <**PATIENT-HAS-COPD**>: El paciente padece EPOC, valores posibles [“YES”, “NO”, “UNDEFINED”](**String**)(**Opcional**)
- <**PATIENT-OXYGEN-SAT**>: Saturación de O2 del paciente al momento de realizarse el estudio (**Decimal**)(**Opcional**)
- <**TEST-RT-PCR**>: Resultados de la prueba RT-PCR al momento de realizarse el estudio [“POSITIVE”, “NEGATIVE”, “UNDEFINED”] (**String**)(**Opcional**)
- <**PATIENT-PO2**>: Presión parcial de O2 del paciente al momento de realizarse el estudio en mmHg (**Decimal**)(**Opcional**)
- <**PATIENT-HEART-RATE**>: Frecuencia cardíaca del paciente al momento de realizarse el estudio en latidos por minuto (**Número entero**)(**Opcional**)
- <**PATIENT-BREATHING-FREQ**>: Frecuencia respiratoria del paciente al momento de realizarse el estudio en respiraciones por minuto (**Número entero**)(**Opcional**)
- <**PATIENT-BLOOD-PRESSURE**>: Presión arterial del paciente al momento de realizarse el estudio en mmHg (**Decimal**)(**Opcional**)

#### Solicitud de ejemplo:

```
curl --location --request POST
  'https://api.cov.x.com/covid-x/training' \
--header 'Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9' \
--form 'chest_x_ray_image=@/image.dcm' \
--form 'birthday=2000-04-12T10:00:00-03:00' \
--form 'sex=MALE' \
--form 'temperature=36.4' \
```

```
—form 'height=1.82' \  
—form 'weight=79.5' \  
—form 'has_asthma=YES' \  
—form 'has_copd=NO' \  
—form 'oxygen_saturation=98.8' \  
—form 'test_rt_pcr=POSITIVE' \  
—form 'partial_pressure_of_oxygen=84' \  
—form 'heart_rate=67' \  
—form 'breathing_frequency=17' \  
—form 'blood_pressure=112.3'
```

**Respuesta de ejemplo:**

```
"1.2.826.0.1.3680043.6.26569.20116.20190403133914"
```

**Código de estado HTTP:** 200

**Respuesta errónea de ejemplo:**

```
{  
  "error_code": "forbidden",  
  "description": "User not authorized",  
  "status": 403  
}
```

**Código de estado HTTP:** 403

### A.3.0.2 Realizar una inferencia

Realiza una inferencia a partir de los datos enviados y retorna la predicción.

**Ruta:** /covid-x/inference

**Acceso:** Privado, Rol requerido: “covid-x-inference-set-writer”

**Método HTTP:** POST

**Encabezados HTTP:** Authorization: Bearer <JWT-TOKEN>

**Campos de formulario de petición HTTP:**

```
– chest_x_ray_image: <DCM-FILE>  
– birthday: <PATIENT-BIRTHDAY>,
```

```

- sex: <PATIENT-SEX>,
- temperature: <PATIENT-TEMPERATURE>,
- height: <PATIENT-HEIGHT>,
- weight: <PATIENT-WEIGHT>,
- has_asthma: <PATIENT-HAS-ASTHMA>,
- has_copd: <PATIENT-HAS-COPD>,
- oxygen_saturation: <PATIENT-OXYGEN-SAT>,
- partial_pressure_of_oxygen: <PATIENT-PO2>,
- heart_rate: <PATIENT-HEART-RATE>,
- breathing_frequency: <PATIENT-BREATHING-FREQ>,
- blood_pressure: <PATIENT-BLOOD-PRESSURE>

```

- <**JWT-TOKEN**>: Token JWT obtenido del login del usuario  
(**String**)(**requerido**)
- <**DCM-FILE**>: Binario del archivo DICOM (Archivo .dcm) (**requerido**)
- <**PATIENT-BIRTHDAY**>: Fecha de nacimiento del paciente  
(**String**)(**RFC3339**)(**Opcional**)
- <**PATIENT-SEX**>: El sexo del paciente, valores posibles [“MALE”,  
“FEMALE”, “OTHER”](**String**)(**Opcional**)
- <**PATIENT-TEMPERATURE**>: La temperatura del paciente al  
momento de realizarse el estudio en grados centígrados  
(**Decimal**)(**Opcional**)
- <**PATIENT-HEIGHT**>: La altura del paciente al momento de re-  
alizarse el estudio en metros (**Decimal**)(**Opcional**)
- <**PATIENT-WEIGHT**>: El peso del paciente al momento de re-  
alizarse el estudio en kilogramos (**Decimal**)(**Opcional**)
- <**PATIENT-HAS-ASTHMA**>: El paciente padece asma, valores  
posibles [“YES”, “NO”, “UNDEFINED”](**String**)(**Opcional**)
- <**PATIENT-HAS-COPD**>: El paciente padece EPOC, valores posi-  
bles [“YES”, “NO”, “UNDEFINED”](**String**)(**Opcional**)
- <**PATIENT-OXYGEN-SAT**>: Saturación de O2 del paciente al  
momento de realizarse el estudio (**Decimal**)(**Opcional**)
- <**PATIENT-PO2**>: Presión parcial de O2 del paciente al momento  
de realizarse el estudio en mmHg (**Decimal**)(**Opcional**)

- <**PATIENT-HEART-RATE**>: Frecuencia cardíaca del paciente al momento de realizarse el estudio en latidos por minuto (**Número entero**)(**Opcional**)
- <**PATIENT-BREATHING-FREQ**>: Frecuencia respiratoria del paciente al momento de realizarse el estudio en respiraciones por minuto (**Número entero**)(**Opcional**)
- <**PATIENT-BLOOD-PRESSURE**>: Presión arterial del paciente al momento de realizarse el estudio en mmHg (**Decimal**)(**Opcional**)

**Solicitud de ejemplo:**

```
curl --location --request POST
  'https://api.cov.x.com/covid-x/inference' \
--header 'Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9' \
--form 'chest_x_ray_image=@/image.dcm' \
--form 'birthday=2000-04-12T10:00:00-03:00' \
--form 'sex=MALE' \
--form 'temperature=36.4' \
--form 'height=1.82' \
--form 'weight=79.5' \
--form 'has_asthma=YES' \
--form 'has_copd=NO' \
--form 'oxygen_saturation=98.8' \
--form 'partial_pressure_of_oxygen=84' \
--form 'heart_rate=67' \
--form 'breathing_frequency=17' \
--form 'blood_pressure=112.3'
```

**Respuesta de ejemplo:**

```
{
  "covid-19": 0.20,
  "pneumonia": 0.64,
  "normal": 0.16
}
```

**Código de estado HTTP: 200**

**Respuesta errónea de ejemplo:**

```
{
  "error_code": "forbidden",
  "description": "User not authorized",
  "status": 403
}
```

**Código de estado HTTP: 403**

### **A.3.0.3 Búsqueda de estudios**

**Ruta:** /covid-x/studies

**Acceso:** Privado, Rol requerido: "covid-x-study-reader"

**Método HTTP:** GET

**Cuerpo de la petición HTTP:**

```
{
  "min_version": <MIN-VERSION>,
  "max_version": <MAX-VERSION>,
  "creation_date_from": <CREATION-DATE-FROM>,
  "creation_date_to": <CREATION-DATE-TO>,
  "birthday_from": <BIRTHDAY-FROM>,
  "birthday_to": <BIRTHDAY-TO>,
  "sex": <PATIENT-SEX>,
  "min_temperature": <MIN-TEMP>,
  "max_temperature": <MAX-TEMP>,
  "min_height": <MIN-HEIGHT>,
  "max_height": <MAX-HEIGHT>,
  "min_weight": <MIN-WEIGHT>,
  "max_weight": <MAX-WEIGHT>,
  "has_asthma": <HAS-ASTHMA>,
  "has_copd": <HAS-COPD>,
  "min_oxygen_saturation": <MIN-OX-SAT>,
  "max_oxygen_saturation": <MAX-OX-SAT>,
  "min_partial_pressure_of_oxygen": <MIN-PO2>,
}
```

```

"max_partial_pressure_of_oxygen": <MAX-PO2>,
"min_heart_rate": <MIN-HEART-RATE>,
"max_heart_rate": <MAX-HEART-RATE>,
"min_breathing_frequency": <MIN-BREATHING-FREQ>,
"max_breathing_frequency": <MAX-BREATHING-FREQ>,
"min_blood_pressure": <MIN-BLOOD-PRESSURE>,
"max_blood_pressure": <MAX-BLOOD-PRESSURE>,
"test_rt_pcr": <RT-PCR-RESULT>
}

```

- <MIN-VERSION> y <MAX-VERSION>: rango de versiones del modelo de estudio de COVID-X  
(Número entero)(requerido)
- <BIRTHDAY-FROM> y <BIRTHDAY-TO>: Rango de fechas de nacimiento del paciente  
(String)(RFC3339)(Opcional)
- <PATIENT-SEX>: El sexo del paciente, valores posibles ["MALE", "FEMALE", "OTHER"] (String)(Opcional)
- <MIN-TEMPERATURE> y <MAX-TEMPERATURE>: Rango de temperatura del paciente en grados centígrados  
(Decimal)(Opcional)
- <MIN-HEIGHT> y <MAX-HEIGHT>: Rango de altura del paciente en metros (Decimal)(Opcional)
- <MIN-WEIGHT> y <MAX-WEIGHT>: Rango de peso del paciente en kilogramos (Decimal)(Opcional)
- <HAS-ASTHMA>: El paciente padece asma, valores posibles ["YES", "NO", "UNDEFINED"] (String)(Opcional)
- <HAS-COPD>: El paciente padece EPOC, valores posibles ["YES", "NO", "UNDEFINED"] (String)(Opcional)
- <MIN-OX-SAT> y <MAX-OX-SAT>: Rango de saturación de O2 del paciente  
(Decimal)(Opcional)
- <MIN-PO2> y <MAX-PO2>: Rango de presión parcial de O2 en mmHg (Decimal)(Opcional)

- **<MIN-HEART-RATE>** y **<MAX-HEART-RATE>**: Rango frecuencia cardíaca del paciente en latidos por minuto (**Número entero**)(**Opcional**)
- **<MIN-BREATHING-FREQ>** y **<MAX-BREATHING-FREQ>**: Rango de frecuencia respiratoria del paciente en respiraciones por minuto (**Número entero**)(**Opcional**)
- **<MIN-BLOOD-PRESSURE>** y **<MAX-BLOOD-PRESSURE>**: Rango de presión arterial del paciente en mmHg (**Decimal**)(**Opcional**)
- **<TEST-RT-PCR>**: Resultados de la prueba RT-PCR ["POSITIVE", "NEGATIVE", "UNDEFINED"] (**String**)(**Opcional**)

**Solicitud de ejemplo:**

```
curl --location --request GET
  'https://api.cov.x.com/covid-x/studies?limit=1' \
--header 'Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9' \
--header 'Content-Type: application/json' \
--data-raw '{
  "min_version": 1,
  "max_version": 1,
  "creation_date_from": "2020-04-12T10:00:00-03:00",
  "creation_date_to": "2020-08-12T10:00:00-03:00",
  "birthday_from": "1951-04-12T10:00:00-03:00",
  "birthday_to": "2002-04-12T10:00:00-03:00",
  "min_temperature": 34.5,
  "max_temperature": 39.5,
  "min_height": 1.45,
  "max_height": 2.00,
  "min_weight": 45,
  "max_weight": 95,
  "min_oxygen_saturation": 90,
  "max_oxygen_saturation": 100,
  "min_partial_pressure_of_oxygen": 75,
  "max_partial_pressure_of_oxygen": 100,
  "min_heart_rate": 60,
```

```
"max_heart_rate": 100,  
"min_breathing_frequency": 12,  
"max_breathing_frequency": 21,  
"min_blood_pressure": 80,  
"max_blood_pressure": 120  
},
```

### Respuesta de ejemplo:

```
{  
  "paging": {  
    "total": 2,  
    "limit": 1,  
    "offset": 0  
  },  
  "result": [  
    {  
      "study_instance_uid":  
        "1.2.826.0.1.3680043.6.26569.20116.2019...",  
      "dcm_url": "https://covid-x-files/raw/1.2.82...",  
      "bmp_url": "https://covid-x-files/con/1.2.82...",  
      "version": 1,  
      "uploader_user_id": 1,  
      "uploader_email": "aaquili@itba.edu.ar",  
      "creation_date": "2020-06-14T23:49:30.185Z",  
      "organization": "itba",  
      "patient_birthday": "2000-04-12T13:00:00Z",  
      "patient_sex": null,  
      "patient_temperature": 36.4,  
      "patient_height": 1.82,  
      "patient_weight": 79.5,  
      "patient_has_asthma": "Yes",  
      "patient_has_copd": "NO",  
      "patient_oxygen_saturation": 98.8,  
      "test_rt_pcr": "POSITIVE",  
      "patient_partial_pressure_of_oxygen": 84,  
      "patient_heart_rate": 67,  
      "patient_breathing_frequency": 17,  
      "patient_blood_pressure": 112.3  
    }  
  ]  
}
```

```
}  
}
```

**Código de estado HTTP:** 200

## A.4 INFERENCE-GATEWAY

### A.4.0.1 Realizar una inferencia

**Ruta:** /predict

**Acceso:** Privado

**Método HTTP:** POST

**Cuerpo de la petición HTTP:**

```
{  
  "chest_x_ray_image_url": <IMAGE-URL>,  
  "clinical_data": {  
    "birthday": <BIRTHDAY>,  
    "sex": <SEX>,  
    "temperature": <TEMPERATURE>,  
    "height": <HEIGHT>,  
    "weight": <WEIGHT>,  
    "has_asthma": <HAS-ASTHMA>,  
    "has_copd": <HAS-COPD>,  
    "oxygen_saturation": <OX-SAT>,  
    "partial_pressure_of_oxygen": <PO2>,  
    "heart_rate": <HEART-RATE>,  
    "breathing_frequency": <BREATHING-FREQ>,  
    "blood_pressure": <BLOOD-PRESSURE>  
  }  
}
```

- **<IMAGE-URL>**: URL a la imagen BMP de la radiografía de tórax (**String**)(**requerido**)
- **<BIRTHDAY>**: Fecha de nacimiento del paciente (**String**)(**RFC3339**)(**Opcional**)
- **<SEX>**: El sexo del paciente, valores posibles [“MALE”, “FEMALE”, “OTHER”](**String**)(**Opcional**)

- <**TEMPERATURE**>: La temperatura del paciente al momento de realizarse el estudio en grados centígrados (**Decimal**)(**Opcional**)
- <**HEIGHT**>: La altura del paciente al momento de realizarse el estudio en metros (**Decimal**)(**Opcional**)
- <**WEIGHT**>: El peso del paciente al momento de realizarse el estudio en kilogramos (**Decimal**)(**Opcional**)
- <**HAS-ASTHMA**>: El paciente padece asma, valores posibles ["YES", "NO", "UNDEFINED"](**String**)(**Opcional**)
- <**HAS-COPD**>: El paciente padece EPOC, valores posibles ["YES", "NO", "UNDEFINED"](**String**)(**Opcional**)
- <**OX-SAT**>: Saturación de O2 del paciente al momento de realizarse el estudio (**Decimal**)(**Opcional**)
- <**PO2**>: Presión parcial de O2 del paciente al momento de realizarse el estudio en mmHg (**Decimal**)(**Opcional**)
- <**HEART-RATE**>: Frecuencia cardíaca del paciente al momento de realizarse el estudio en latidos por minuto (**Número entero**)(**Opcional**)
- <**BREATHING-FREQ**>: Frecuencia respiratoria del paciente al momento de realizarse el estudio en respiraciones por minuto (**Número entero**)(**Opcional**)
- <**BLOOD-PRESSURE**>: Presión arterial del paciente al momento de realizarse el estudio en mmHg (**Decimal**)(**Opcional**)

#### Solicitud de ejemplo:

```
curl --location --request POST
  'https://ai-proxy.cov-x.com/predict' \
--header 'Content-Type: application/json' \
--data-raw '{
  "chest_x_ray_image_url": "https://cov-x-files/1.bmp",
  "clinical_data": {
    "birthday": "2000-04-12T10:00:00-03:00",
    "sex": "MALE",
    "temperature": 36.4,
```

```
"height": 1.82,  
"weight": 79.5,  
"has_asthma": "YES",  
"has_copd": "NO",  
"oxygen_saturation": 98.8,  
"partial_pressure_of_oxygen": 84,  
"heart_rate": 67,  
"breathing_frequency": 17,  
"blood_pressure": 112.3  
}  
'
```

**Respuesta de ejemplo:**

```
{  
  "covid-19": 0.20,  
  "pneumonia": 0.64,  
  "normal": 0.16  
}
```

**Código de estado HTTP:** 200