

# **PROYECTO FINAL DE INGENIERÍA**

## **StopFire: ALERTA DE INCENDIOS FORESTALES EN ARGENTINA USANDO IoT Y MACHINE LEARNING**

**Alric, Juan Cruz – LU 1097731**

Ingeniería en Informática

**Curbelo, Alejandra – LU 1110438**

Ingeniería en Informática

Tutor:

**Inchausti, Pablo Ezequiel, UADE**

**2022**

# **UADE**

**UNIVERSIDAD ARGENTINA DE LA EMPRESA**

**FACULTAD DE INGENIERÍA Y CIENCIAS EXACTAS**

## Agradecimientos

A mi mujer Jacinta, por estar siempre desde que decidí cambiarme de carrera y seguir con Ingeniería Informática.

A mi Padre que me apoyó en este cambio y me ayudó un montón.

A Alejandra, mi compañera de tesis, que sin ella esto no se podría haber concretado.

A Pablo, nuestro tutor, que nos estuvo acompañando y ayudando en este arduo camino.

A mis amigos de la carrera, que compartimos muchas materias juntos y nos apoyamos mutuamente para poder llegar hasta acá.

Juan Cruz

A mi novio Fernando, por acompañarme durante mi paso por la UADE, por darme su apoyo cada vez que lo necesito y por transmitirme la calma y tranquilidad que muchas veces me es difícil conseguir en mí misma.

A mi mamá por ser mi fan #1 en todo lo que hago y por estar siempre presente sin importar la distancia.

A todos lo que aportaron su granito de arena para que hoy me pudiera recibir:

- Mi papá que me apoyó cuando decidí mudarme a la Argentina.
- Mis hermanos y en especial mi hermanito Juanmi quien fue mi compañero de inmigración y la única persona que he tenido físicamente a mi lado durante 26 años de mi vida. Eres el mejor compañero del mundo, te amo.
- Mis profesores de LUZ, de la UBA y de UADE por compartir su conocimiento y su pasión por enseñar. Mención especial a: Francisco Luengo, Sigerist Rodríguez, María Emilia Portillo, Gerardo Pirela, Olinto Rodríguez, Yaskelly Yedra, Ricardo Sirne, Claudia López, Marcelo Fontana, Josefina Silveyra, Fernando Herrera y Joselyn Camero.
- Los Briceño Moran por ser mi familia en Argentina, ¡espero a partir de ahora no perderme ninguna fiesta más por tener que quedarme estudiando!
- Mis amigas que me hicieron el camino más llevadero y divertido: Ale, Dani y Vane.
- Mis compañeros durante largas jornadas de estudio: Daniela, Gustavo y Molano.
- Mis mentores profesionales y de la vida: Danny, Nico y Adrián, a quienes recurro cuando necesito orientación y que siempre están disponibles para mí.

A nuestro tutor, Pablo. Gracias por tu acompañamiento y guía desde el día uno. Nos ayudaste a entregar un trabajo de excelente calidad y del cual nos sentimos super orgullosos.

A mi compañero de tesis, Juan. Creo que cuando se trabaja en equipo es muy importante poder complementarse y siento que armamos una dupla muy balanceada donde mis falencias fueron compensadas por tus skills, así que gracias por tu compromiso desde el principio. ¡Lo logramos!

A todos ellos, muchísimas gracias.

Alejandra

## Resumen

Los incendios forestales suceden no solo en la Argentina, sino que en el mundo entero. Su consecuencia más grave es que acaban con la biodiversidad de la zona y a la naturaleza le toma decenas de años poder recuperarse de este tipo de desastres.

En los últimos 2 años, según el Servicio Nacional de Manejo del Fuego, la superficie estimada afectada por incendios reportados entre el 1 de enero del 2020 y el 2 de diciembre de 2021 en todo el territorio nacional es de 1 millón 453 mil hectáreas. Lo que equivale a 71 veces la superficie de Capital Federal (Servicio Nacional de Manejo del Fuego. 2021).

Este trabajo de investigación propone combinar internet de las cosas (IoT) con machine learning con el objetivo de monitorear y detectar incendios forestales capturados por cámaras.

Se desarrolló un modelo de machine learning para la detección de los incendios a través del análisis de imágenes (computer vision). A su vez se propuso una aplicación web para poder visualizar en un mapa los incendios detectados por la red de dispositivos IoT y una aplicación móvil para patrulleros en la zona que deseen enviar una alerta, compartiendo no solo la imagen del incendio o del humo, sino que también su geolocalización para facilitarle a los entes responsables el poder identificar la ubicación geográfica del desastre de manera inmediata.

**Abstract**

Forest fires occur not only in Argentina but throughout the world. Its biggest impact is that it destroys the biodiversity of the area, and it takes nature dozens of years to recover from this type of disaster.

In the last 2 years, according to the National Fire Management Service, the estimated area affected by fires reported between January 1<sup>st</sup> 2020, and December 2<sup>nd</sup> 2021, throughout the national territory is 1 million 453 thousand hectares. This is equivalent to 71 times the area of the Capital Federal (Servicio Nacional de Manejo del Fuego. 2021).

This research work proposes to combine the Internet of Things (IoT) with machine learning to monitor and detect forest fires captured by cameras. The goal was to develop a machine learning model to detect forest fires implementing computer vision. Also, develop a web application to be able to visualize on a map the fires detected by the cameras that belong to the IoT network and a mobile application for patrolmen in the area who want to send an alert, sharing not only the image of the fire or smoke but also its geolocation to simplify the job of the responsible entities to identify the location of the disaster immediately.

**Tabla de contenidos**

**1. Introducción..... 9**

1.1. Objetivos..... 9

1.2. Alcance..... 9

**2. Antecedentes ..... 11**

2.1. Marco teórico..... 11

2.1.1. Incendios forestales en Argentina ..... 11

2.1.1.1. Factores ambientales ..... 12

2.1.1.2. Causas antrópicas ..... 14

2.1.2. Machine learning..... 15

2.1.2.1. Redes neuronales ..... 15

2.1.2.2. Computer vision ..... 16

2.1.2.3. Redes neuronales convolucionales ..... 17

2.1.2.4. Evaluación de modelos..... 20

2.1.3. Internet of Things (IoT)..... 22

2.1.3.1. Servicios de IoT en Cloud Providers..... 23

2.1.3.2. Protocolos de conectividad para IoT ..... 24

2.2. Estado del arte ..... 30

2.2.1. Proyectos existentes para detectar incendios con ML..... 30

2.2.2. Algoritmos de reconocimiento de imágenes ..... 31

2.3. User Research..... 32

2.3.1. Entrevistas ..... 32

**3. Descripción ..... 33**

---

3.1.	Desarrollo de una red neuronal para detección de incendios a través de imágenes .....	33
3.1.1.	Set de datos a utilizar.....	33
3.1.2.	Preparación de los datos .....	34
3.1.3.	Entrenamiento y evaluación del modelo .....	36
3.1.3.1.	Proceso de selección del learning rate.....	37
3.1.3.2.	Entrenamiento y evaluación de la CNN .....	37
3.1.4.	Conclusiones.....	44
3.2.	Dashboard de alerta de incendios .....	44
3.2.1.	Mockups .....	45
3.3.	Aplicación móvil .....	47
3.3.1.	Wireframes .....	48
3.4.	Arquitectura.....	51
3.4.1.	Diagrama de arquitectura .....	51
3.4.2.	Diagrama de base de datos .....	53
3.5.	Dispositivos IoT para la detección temprana de incendios forestales .....	54
3.5.1.	Dispositivos existentes .....	56
3.5.1.1.	Definición y características .....	56
3.5.1.2.	Autonomía del dispositivo en el campo .....	66
3.5.1.3.	Protocolos de comunicación.....	68
3.5.1.4.	Tipos de procesamiento .....	70
3.5.1.5.	Costos de implementación.....	73
3.5.2.	Integración con los principales Cloud Providers en el mercado .....	76
3.5.2.1.	Amazon Web Services .....	77
3.5.2.2.	Microsoft Azure.....	78

---

3.5.2.3.	Google Cloud.....	79
3.5.3.	Conclusiones.....	80
<b>4.</b>	<b>Metodología y desarrollo .....</b>	<b>83</b>
4.1.	Herramientas utilizadas .....	84
4.2.	Frameworks, lenguajes, algoritmos y librerías.....	85
<b>5.</b>	<b>Pruebas realizadas.....</b>	<b>86</b>
5.1.	Red neuronal convolucional .....	86
5.2.	Envío de imagen y geolocalización desde la aplicación móvil .....	91
<b>6.</b>	<b>Branding.....</b>	<b>96</b>
6.1.	Paleta de colores .....	96
6.2.	Logo.....	97
<b>7.</b>	<b>Plan de negocio .....</b>	<b>98</b>
7.1.	Costos de desarrollo .....	98
7.2.	Modelo de negocio .....	99
<b>8.</b>	<b>Discusión.....</b>	<b>101</b>
<b>9.</b>	<b>Conclusiones.....</b>	<b>102</b>
9.1.	Futuras líneas de investigación.....	103
<b>10.</b>	<b>Bibliografía.....</b>	<b>104</b>
<b>11.</b>	<b>Anexos.....</b>	<b>109</b>
11.1.	Glosario .....	109
11.2.	Entrevistas realizadas .....	111
11.3.	Tabla de figuras .....	115
11.4.	Paper aceptado en CACIC 2022.....	119
11.5.	Cronograma .....	124

## 1. Introducción

### 1.1. Objetivos

El objetivo general de este proyecto final de ingeniería (PFI) es desarrollar un sistema de monitoreo y detección de incendios forestales en la Argentina a través de análisis de imágenes capturadas por dispositivos de IoT en el año 2022.

Son objetivos específicos de este proyecto:

- Desarrollar un modelo de machine learning para analizar imágenes y determinar la existencia de un potencial incendio forestal.
- Investigar una forma adecuada de implementar una red de dispositivos IoT para monitorear zonas propensas a incendios forestales.

En el contexto de este PFI, se entiende por ‘una implementación adecuada’ aquella donde se analizan los requisitos físicos y técnicos para lograr el correcto funcionamiento de las cámaras de monitoreo en zonas áridas e inhóspitas.

- Investigar soluciones de integración de los dispositivos de IoT con los principales Cloud Providers.

Este trabajo de investigación se integra al proyecto de investigación del Instituto de Tecnología (INTEC) de la Universidad Argentina de la Empresa (UADE) llamado “A21T03 – Aplicaciones de Machine Learning para mejorar el uso de Recursos Naturales”.

A su vez, es continuación del proyecto final de ingeniería “AQUA: Desarrollo de un modelo de machine learning para prevenir incendios forestales en Pinamar” (Martínez Saucedo, 2021).

### 1.2. Alcance

El sistema StopFire está conformado por una aplicación web y una aplicación móvil compatible con el sistema operativo Android en idioma español. Las funcionalidades del sistema en su primera versión (Release 1) se explican a continuación.

La aplicación web cuenta con los siguientes módulos:

- Módulo de detección: Se desarrolló un modelo de machine learning para analizar imágenes de zonas forestales detectadas por dispositivos IoT.

Estos dispositivos IoT deben estar conectados a un Cloud Provider (AWS) que va a recibir las imágenes cada 30 minutos, procesarlas y generar una alerta en caso de que el modelo previamente entrenado detecte que existe un posible incendio en la zona.

Las imágenes pueden corresponder a puntos fijos o móviles; por ejemplo, obtenidas desde drones.

Para este MVP, se utilizó la cámara del teléfono celular para simular las cámaras que enviarán las imágenes al servidor.

- Dashboard donde se pueden ver las cámaras conectadas a la red junto a su ubicación geográfica.
- Módulo de alertas donde aparecen las notificaciones de los incendios detectados junto con la foto que el modelo usó para identificar el potencial foco del incendio, la localización geográfica de la cámara que lo detectó y la hora a la que se analizó la imagen que disparó dicha alerta.

En caso de que el incendio provenga de la aplicación móvil, se va a mostrar la latitud y longitud de la ubicación donde estaba el usuario al momento de realizar la notificación del potencial incendio.

La aplicación móvil busca crear una red de personas interesadas en esta problemática (usuarios que deseen hacer patrullajes voluntarios, por ejemplo) y está compuesta por un módulo de avisos a través del cual el usuario puede informar un incendio que esté ocurriendo en el momento, adjuntando una imagen.

Dicha imagen va a ser analizada por el modelo y, si corresponde a un incendio, va a mandar la alerta de la misma forma como si lo hubiese detectado uno de los dispositivos IoT conectados a la red.

Esto les sirve a los bomberos para identificar incendios que ocurran en zonas donde la cámara no tiene visibilidad, así como ‘marcar’ los lugares donde es ideal ampliar la red de dispositivos.

## 2. Antecedentes

Se llevó a cabo un proceso de investigación para entender cuál es el contexto y dimensión de la problemática seleccionada, antecedentes en la Argentina relacionados con el tema, así como precedentes del pilar de tecnología asociado al proyecto: qué es y cómo funcionan las redes neuronales, dispositivos IoT para detección de incendios, conexión de dispositivos IoT a los Cloud Providers existentes en el mercado.

### 2.1. Marco teórico

En el marco teórico se detallan las bases conceptuales que dan sustento a la propuesta de solución de este PFI. Se describirá primero la problemática de los incendios forestales y a continuación las tecnologías asociadas a Machine Learning e IoT utilizadas en el desarrollo del trabajo.

#### 2.1.1. Incendios forestales en Argentina

El problema está relacionado con los incendios forestales ocurridos en el territorio de la República Argentina.

Se conoce como incendio forestal a un fuego que se propaga de manera descontrolada y muy rápida a través de la vegetación de la zona, generando un efecto no deseado en el área donde se expande.

Las causas que originan un incendio forestal se dividen en dos:

- **Casas naturales:** se da cuando el incendio es generado por un factor de la naturaleza. Ejemplo: la caída de un rayo por tormentas eléctricas, la explosión de un volcán en la zona, falta de precipitaciones, altas temperaturas en la zona, etc.
- **Causas antrópicas:** como su nombre lo indica, son producto de la actividad humana. Esto se da gracias a negligencia humana o por fuegos intencionados que se salen de control.

El 95% de los incendios forestales en la Argentina son causados por actividades humanas (Ministerio de Ambiente y Desarrollo Sostenible. 2021).

Según el reporte diario del Servicio Nacional de Manejo del Fuego (SNMF) del día 31 de diciembre del 2021, se estima que las superficies afectadas por incendios reportados en el año 2021 alcanzan las 302 mil hectáreas, siendo la provincia de San Luis, con casi 50 mil hectáreas, la más afectada (Ministerio de Ambiente y Desarrollo Sostenible, 2021).

Este problema que sigue vigente en el 2022. Solamente en el mes de febrero se produjeron incendios en la provincia de Corrientes que alcanzaron las 934 mil hectáreas de acuerdo con el informe proveniente del Instituto Nacional de Tecnología Agropecuaria (INTA) del día 24 de febrero de 2022. Esto equivale al 11% de la superficie total de la provincia (Kurtz y otros. 2022).

#### ***2.1.1.1. Factores ambientales***

Según el SNMF (2018) las variables del comportamiento del fuego son:

- La velocidad de propagación
- La longitud y altura de llama
- La intensidad
- La distancia de propagación
- El área, perímetro y forma
- La ocurrencia de focos secundarios
- El coronamiento en incendios de bosques

Los factores ambientales que influyen en el comportamiento del fuego se resumen en tres grupos: combustible, meteorología y topografía. Esto es conocido como el triángulo del comportamiento del fuego (Servicio Nacional de Manejo del Fuego. 2018).

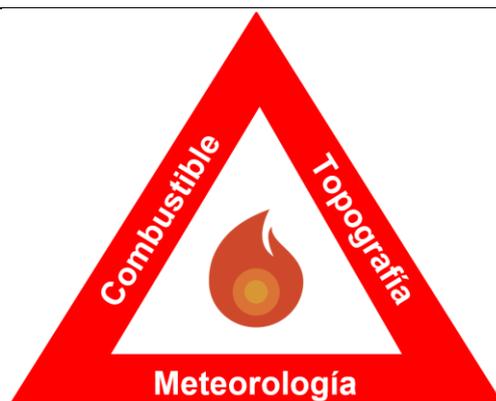


Figura 1: representación del triángulo del comportamiento del fuego

Fuente: elaboración propia

Para combatir el fuego hay que analizar cuidadosamente esos tres aspectos, ya que, al identificar correctamente las características visibles del incendio, se pueden tomar las decisiones adecuadas para poder combatirlo.

### **Meteorología**

Las variables meteorológicas que influyen en la propagación del fuego son: temperatura, humedad, precipitación y viento. Estas van a estar en constante variación ya que dependen del momento del día y de la época del año en que se esté; también van a influir tanto en la propagación del fuego como en el rendimiento de las personas que trabajan para apagarlo (Servicio Nacional de Manejo del Fuego. 2018):

- **Temperatura:** la temperatura del ambiente va a influir directamente en la intensidad, velocidad, dirección de propagación y altura de las llamas. En lugares con temperaturas más alta ( $> 30\text{ }^{\circ}\text{C}$ ) es más probable que el fuego se torne más difícil de controlar.
- **Humedad relativa:** cuanto menor sea la humedad relativa en el ambiente, la humedad en el combustible que esté alimentando el fuego va a ser menor, lo que hará que sea más difícil su extinción.
- **Precipitación:** cuando la cantidad de precipitación en la zona es alta ( $>15\text{mm/hora}$ ), la probabilidad de que haya reservas de agua para apagar el fuego son mayores lo que favorecerá que se extinga el fuego más rápidamente. Mientras que el caso opuesto

---

(cuando se está en época de sequía) crea condiciones favorables para el desarrollo de incendios forestales, debido al aumento de la disponibilidad de combustible en la zona.

- Viento: el viento influye de manera directa la velocidad y dirección del fuego. Una zona con ventiscas muy intensas puede aumentar la cantidad de oxígeno en el aire lo que favorece la expansión del fuego.

### **Topografía**

Se refiere a las variaciones en el suelo donde está ocurriendo el incendio.

Factores como la inclinación del terreno influyen en la velocidad y dirección del fuego, ya que una superficie con una pendiente intensa va a favorecer la velocidad de propagación del fuego en el sentido de dicha pendiente.

La topografía de la zona también puede llegar a complicar la tarea de los bomberos y entes responsables, debido a que zonas con mucha variación en el terreno hacen que sea difícil para las personas trabajar en la extinción de las llamas.

### **Combustible**

Es el principal elemento de la combustión y, en general, se puede encontrar en estado sólido, líquido o gaseoso. En un incendio forestal, el combustible suele ser la misma vegetación de la zona, con la excepción de los casos donde hay incendios intencionales causados por intervención humana donde hay alcohol, gasolina o cualquier combustible que no se encuentra naturalmente en ese bosque y/o suelo.

Cuando se está en presencia de un incendio forestal, hay que buscar eliminar el combustible, porque sin combustible no hay fuego. Una manera de lograrlo es cortando la maleza de zonas aledañas a donde se está produciendo el fuego. De esta forma, cuando las llamas alcancen esa zona, no tendrán de donde alimentarse y se van a extinguir.

#### ***2.1.1.2. Causas antrópicas***

La mayoría de los incendios forestales en la Argentina ocurren como resultado de la actividad humana (ya sea de manera intencional o por negligencia).

Esto se debe a que la presencia del hombre en bosques implica la presencia de elementos que pueden ocasionar un incendio. Algunos ejemplos de situaciones que pueden originar un incendio forestal son:

- Fogatas
- Colillas de cigarrillo mal apagadas
- Quema de basura
- Quema de maleza en la zona
- Vandalismo; ocultar o disimular delitos
- Cacería

### **2.1.2. Machine learning**

En el presente trabajo se recurre a las tecnologías de Machine Learning, y en particular las redes neuronales convolucionales (CNN) para la detección de incendios forestales. Las CNN se usaron en este PFI para analizar y detectar patrones de fuego y humo en las imágenes obtenidas por medio del monitoreo.

#### **2.1.2.1. Redes neuronales**

Las redes neuronales artificiales (RNA) son modelos matemáticos que simulan el comportamiento biológico de las neuronas de los seres humanos (es una abstracción matemática).

El objetivo de una RNA es solucionar problemas complejos (de tipo no lineal) a través de la detección de patrones en un conjunto de datos de entrada.

La unidad mínima de una RNA es el perceptrón (neurona) y la red neuronal es definida como un conjunto de perceptrones interconectados. Cada perceptrón se encarga de una única tarea, lo que hace que el modelo sea más complejo y potente (porque tiene más flexibilidad y más potencia para detectar los patrones).

Los elementos básicos que componen una red neuronal son:

- Capa de entrada
- Capas ocultas
- Capa de salida

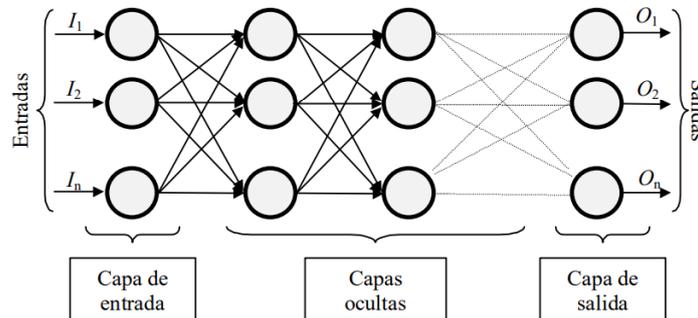


Figura 2: estructura de una red neuronal totalmente conectada

Fuente: (Matich. 2001)

Los datos ingresan por la capa de entrada, pasan a través de las capas ocultas donde se les aplica alguna “transformación” para después salir por la última capa (la capa de salida) con un resultado.

Dichas transformaciones lo que buscan es mejorar el aprendizaje de la red para que pueda identificar patrones relevantes independientemente de cómo vengan los datos de entrada.

### 2.1.2.2. Computer vision

Al momento de analizar imágenes de manera automatizada, entra en juego la ‘visión por computadora’ (más conocido como computer vision) que es “un campo de estudio enfocado en el análisis de imágenes del mundo real a través de algoritmos especializados” (Szeliski. 2011). En otras palabras, es la disciplina que busca ‘ayudar a que la computadora vea’.

Algunos ejemplos de aplicaciones reales son (Szeliski. 2011):

- Reconocimiento óptico de caracteres: reconocimiento de patentes en la vía pública
- Inspección de máquinas: detectar defectos en equipos mecánicos a través del uso de rayos-x y análisis de dichas imágenes.
- Análisis de imágenes médicas para la detección de anomalías.
- Seguridad automotriz: detectar obstáculos inesperados en la vía pública por autos autónomos.
- Captura de movimiento: usar marcadores retro reflectantes vistos desde múltiples cámaras para capturar actores para animación por computadora;

- Vigilancia: detección de intrusos en propiedad privada.
- Reconocimiento de huellas dactilares y biometría con aplicaciones forenses.

Como se puede inferir con la variedad de ejemplos anteriores, computer vision se puede aplicar desde muchos enfoques diferentes. Algunos de estos pueden ser:

- Clasificación de objetos: ¿a qué clase pertenece?
- Identificación de objetos: ¿qué tipo de objeto es?
- Verificación de objetos: ¿está en la imagen X objeto?
- Detección de objetos: ¿dónde está en la imagen?
- Detección de landmarks: ¿cuáles son los puntos característicos del objeto?
- Segmentación de objetos: ¿qué píxeles pertenecen al objeto?
- Reconocimiento de objetos: ¿qué objetos hay en la imagen y dónde están? detección de objetos + identificación de objetos

### ***2.1.2.3. Redes neuronales convolucionales***

Ahora que ya se explicó que es una red neuronal y cuál es la disciplina especializada en el análisis de imágenes a través de computadoras, se puede hablar de las redes neuronales convolucionales (CNN, por sus siglas en inglés).

La red neuronal convolucional (CNN) es un tipo especial de red neuronal multicapa o arquitectura de aprendizaje profundo inspirada en el sistema visual de los seres vivos. La CNN es muy adecuada para diferentes campos de la visión artificial y el procesamiento del lenguaje natural.

Las CNN son un tipo de neurona artificial que tienen una arquitectura feedforward (los modelos feedforward son aquellos modelos donde las neuronas de una capa no se conectan con neuronas de la misma capa, pero sí con todas las de la capa siguiente), lo que le da una gran habilidad para aprender características muy abstractas de los objetos y permite identificar patrones de manera más eficiente a la hora de poner en práctica dicho modelo.

Estas redes han demostrado tener un excelente rendimiento para abordar y solucionar muchos problemas de computer vision y machine learning (aprendizaje automático).

La arquitectura de una CNN consiste en un conjunto finito de capas de procesamiento que pueden aprender varias características de los datos de entrada (es decir, de las imágenes) con múltiples niveles de abstracción.

Las primeras capas de la red aprenden y extraen las características de alto nivel (con menor abstracción), y las capas más profundas aprenden y extraen las características de bajo nivel (con mayor abstracción).

Como se puede observar en la figura 3 citada del paper “Fundamental Concepts of Convolutional Neural Network” (Ghosh y otros. 2020), las CNN constan de 3 tipos de capas intermedias:

- Capa convolucional (convolutional layer): es el componente más importante de cualquier arquitectura CNN. Contiene un conjunto de núcleos convolucionales (también llamados filtros), que se convolucionan con la imagen de entrada para generar un mapa de características de salida.
- Capa de agrupación (pooling layer): esta capa se usa para sub-muestrear los mapas de características (producidos después de las operaciones de convolución); es decir, toma los mapas de características de mayor tamaño y los reduce a mapas de características de menor tamaño.
- Capas completamente conectadas (fully connected layer): estas capas son las que se usan como clasificador de la arquitectura, Toman como entrada la información de la última capa de agrupación (que es la que contiene el mapa de características final), crea un vector con esas métricas y luego lo pasa por la red feedforward para generar la salida. En la figura 3, la red feedforward que procesa el vector final es la llamada “fully connected layer”.

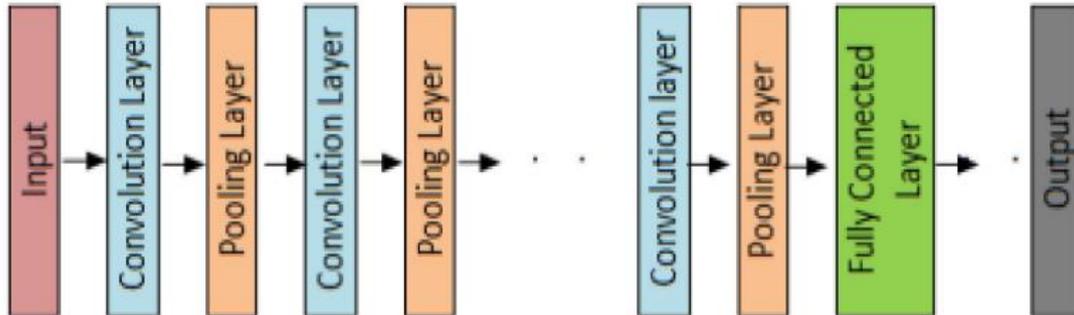


Figura 3: modelo básico conceptual de una CNN

Fuente: (Ghosh y otros. 2020)

Lo descrito anteriormente es la explicación conceptual de una CNN. Ahora, al momento de analizar una imagen, se pueden enumerar los siguientes pasos:

1. Se recibe la imagen a analizar
2. Se procesa la imagen
3. Se extraen características relevantes
4. Se detectan el objeto de interés
5. Se clasifica el objeto

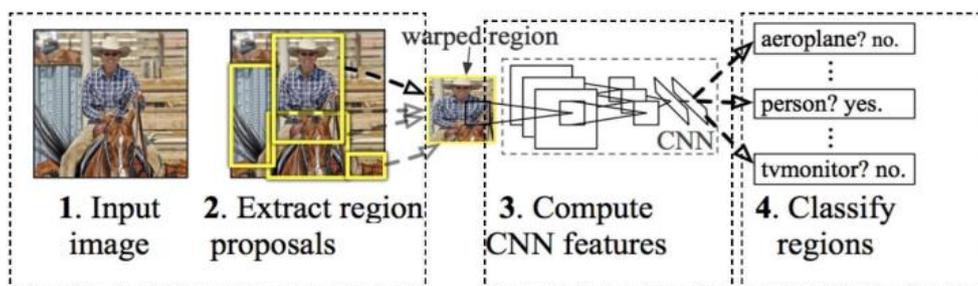


Figura 4: arquitectura de una CNN

Fuente: (Ghosh y otros. 2020)

#### 2.1.2.4. Evaluación de modelos

Después de desarrollada la CNN, se debe evaluar el rendimiento para determinar su exactitud prediciendo a qué categoría pertenece una imagen dada.

Hay distintas maneras de evaluar la efectividad de una CNN. Cuando se habla de predicciones binarias, se suele utilizar la matriz de confusión.

La matriz de confusión sirve para graficar la cantidad de predicciones correctas (verdaderos negativos (TN) y verdaderos positivos (TP)) en comparación con la cantidad de predicciones erróneas (falsos positivos (FP) y falsos negativos (FN)).

Como se observa en la figura 5, la diagonal principal contiene la suma de todas las predicciones correctas y la diagonal inversa contiene las predicciones incorrectas.

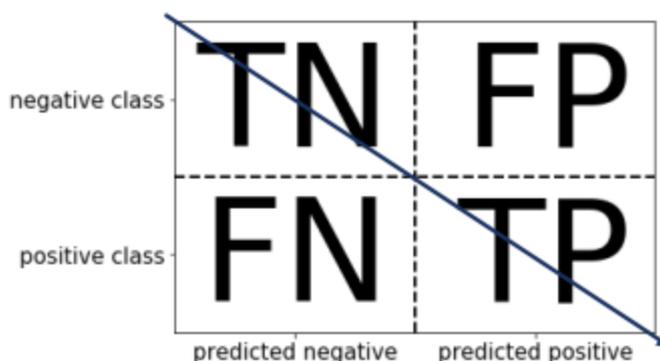


Figura 5: matriz de confusión de un modelo de clasificación binario

Fuente: elaboración propia

TN (true negative o verdaderos negativos): clase negativa que se predijo como negativa.

TP (true positive o verdaderos positivos): clase positiva que se predijo como positiva.

FN (false negative o falsos negativos): clase positiva que se predijo como negativa.

FP (false positive o falsos positivos): clase negativa que se predijo como positiva.

De la matriz de confusión se pueden sacar las siguientes métricas:

- Exactitud (o accuracy, en inglés): representa el porcentaje de predicciones correctas. Sirve para entender el rendimiento del modelo en general.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- Precisión (precision, en inglés): representa el porcentaje de predicciones positivas correctas en comparación con el total de predicciones positivas.

Un escenario donde se puede utilizar es en una campaña de Marketing, porque si se desea enviar una invitación de un evento, no se quiere dejar por fuera a las personas con alta chance de asistir, por lo que es mejor invitar a todos los contactos que dejar por fuera a muchos potenciales asistentes.

$$precision = \frac{TP}{FP + TP}$$

- Sensibilidad (recall, en inglés): representa el porcentaje de casos positivos detectados en comparación a todos los positivos existentes.

Se suele utilizar en el campo de la medicina, ya que es ‘menos dañino’ predecir que un paciente sano está enfermo (falso positivo) que decir que un paciente enfermo está sano (falso negativo).

$$recall = \frac{TP}{FN + TP}$$

### 2.1.3. Internet of Things (IoT)

El internet de las cosas (IoT, por sus siglas en inglés) se refiere a una “red de objetos físicos que están equipados con sensores, software y otras tecnologías con el fin de conectarse e intercambiar datos con otros dispositivos y sistemas a través de internet” (Oracle. 2019).

Estos dispositivos pueden ser objetos comunes (por ejemplo, relojes inteligentes) o máquinas más complejas (autos eléctricos autónomos).

El dispositivo IoT en este proyecto de investigación son las cámaras.

En la Argentina actualmente se pueden conseguir cámaras IP fijas con detectores térmicos integrados que permiten la detección temprana de incendios. Algunas de las cámaras que se pueden adquirir en el país son: cámaras térmicas marca *Hikvision* y *Bosch*.

Según la descripción provista por el proveedor en su página web oficial. las cámaras térmicas de seguridad de *Hikvision* utilizan detectores térmicos avanzados que proporcionan un excelente rendimiento de imagen con alta eficiencia (Hikvision. 2022).

Estas cámaras son ideales para uso interior y requieren estar en un ambiente con protección del viento. Algunos lugares donde es recomendable utilizar estas cámaras son: galpones cerrados, parques industriales, escuelas y supermercados.

Por otro lado, las cámaras térmicas *Bosch* utilizan la tecnología AVIOTEC. “Estos dispositivos cuentan con sensores y analítica de vídeo integrado para detectar llamas y humo directamente en el origen. Por tanto, el dispositivo puede detectar los incendios más rápidamente que un detector común” (Bosch. 2021).

Como lo indica el proveedor, estas cámaras son recomendadas para lugares cerrados o semiabiertos. A pesar de que la detección de incendios en espacios semiabiertos es difícil debido a las influencias del viento, AVIOTEC ofrece detección de humo y llamas incluso en circunstancias meteorológicas desfavorables, gracias al algoritmo de inteligencia artificial que tienen instalado.

Si bien las cámaras *Hikvision* y *Bosch* pueden detectar fuego y humo, no son dispositivos especializados para la detección de incendios en ambientes abiertos, por lo que no son adecuadas para el campo de enfoque de este PFI que son los incendios forestales. Además de que tienen un costo 9 veces mayor que el de una cámara de IP.

### *2.1.3.1. Servicios de IoT en Cloud Providers*

#### **Cloud Computing**

La computación en la nube (mejor conocida como Cloud) se refiere al enfoque que busca “mover” la infraestructura IT a la nube, permitiendo acceder a los datos a través de internet, sin necesidad de que las empresas deban depender de servidores instalados localmente en sus oficinas.

Esta nube va a constar de una red de servidores físicos que es gestionada por un proveedor de servicio (mejor conocido como Cloud Provider), quien es el encargado del mantenimiento de la infraestructura física (hardware).

Esto les permite a las empresas:

- Reducir costos de IT, ya que tercerizan los costos de compra, instalación, soporte y mantenimiento de servidores físicos
- Mayor escalabilidad porque se puede ir aumentando o disminuyendo la cantidad de recursos que deseen utilizar según la demanda.
- Mayor flexibilidad a la hora de acceder a los servicios internos de la empresa, puesto que solo se necesita una conexión a internet.

### Cloud providers que lideran el mercado

Los principales proveedores de servicios en la nube son: Microsoft, Amazon, y Google. Estos proveedores ofrecen distintos servicios dirigidos a IoT que se pueden ver en la figura 6.

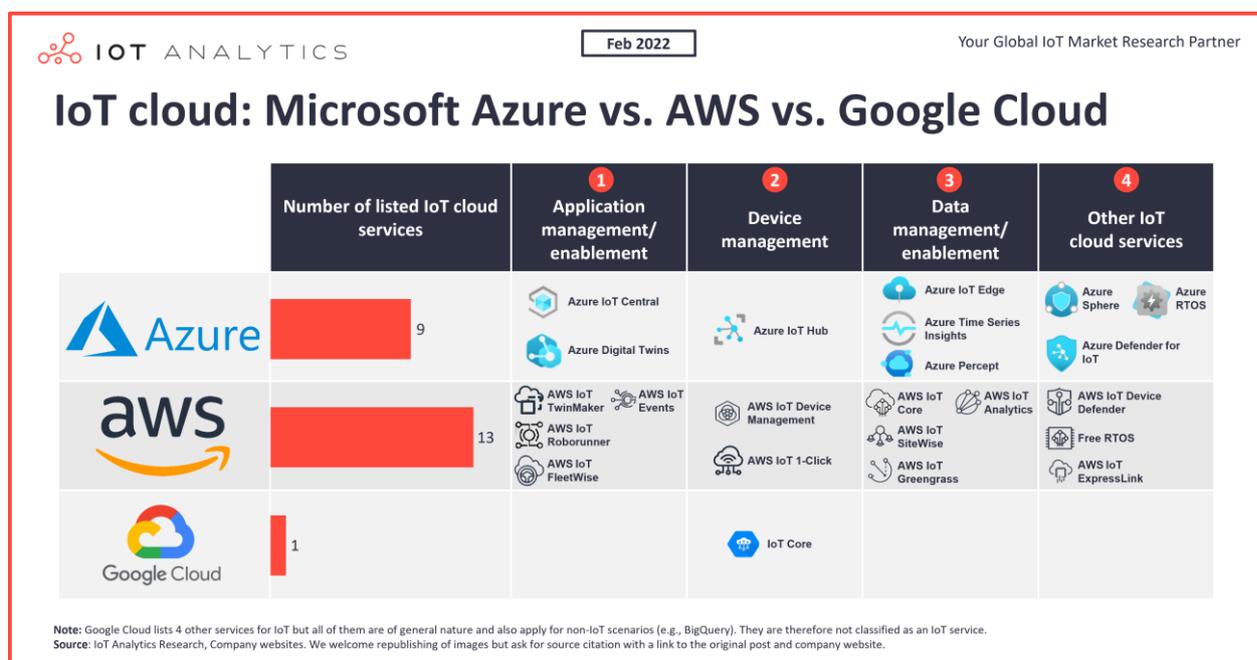


Figura 6: servicios IoT en la nube por proveedor

Fuente: (Hasan. 2022)

#### 2.1.3.2. Protocolos de conectividad para IoT

Antes de hablar de la conectividad, se debe recordar el modelo OSI para telecomunicaciones.

Este modelo está conformado por 7 capas. Cada capa da servicios a la capa superior y establece una comunicación con la misma capa de otro dispositivo.

La información se divide en PDUs (Protocol Data Units, en inglés) y cada capa envía sus PDU a la capa inferior, quien agrega un encabezado para identificar origen y destino. Este proceso se llama encapsulamiento.

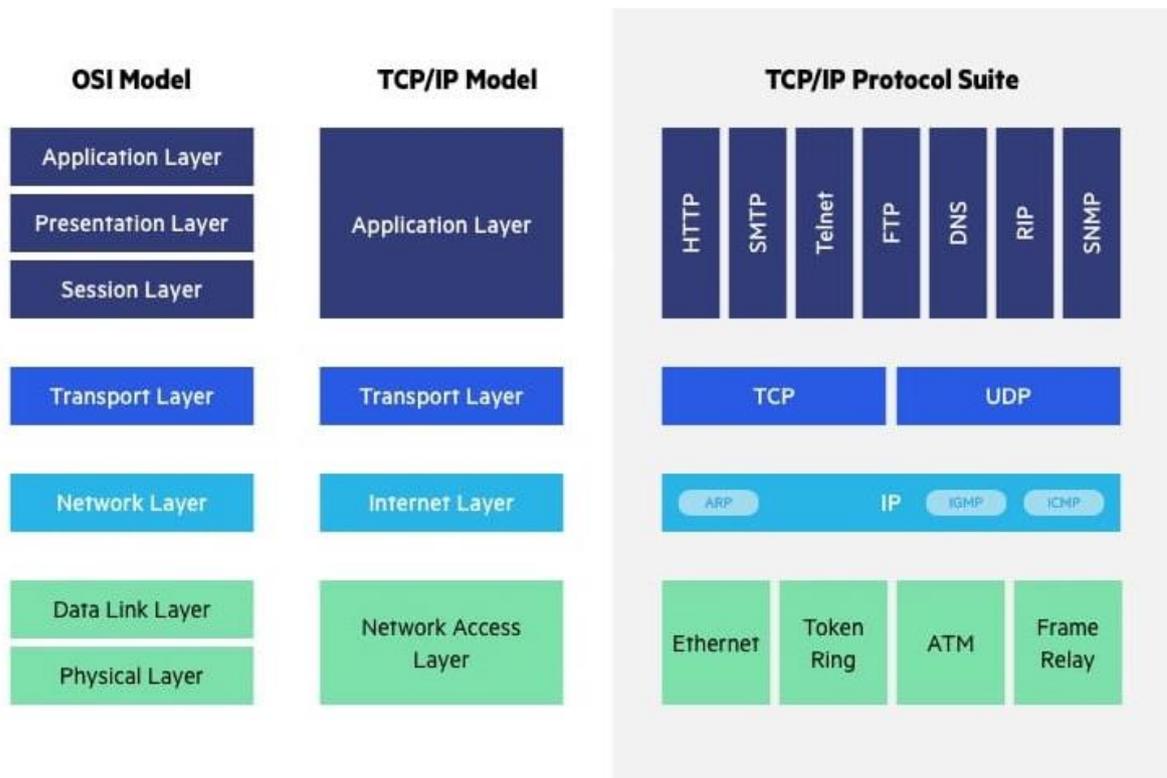


Figura 7: capas del modelo OSI

Fuente: (Imperva. 2021)

Ahora bien, para hablar sobre conectividad IoT, se debe empezar por la capa de transporte que es la encargada de transportar los datos entre hosts, así como proveer servicios de control de flujo, retransmisión y control de errores.

Es importante enfocarse en esta capa ya que, para lo que concierne este PFI, la comunicación entre el dispositivo IoT y el Cloud Provider son de suma importancia, porque de esto depende que el modelo pueda analizar la imagen y, de ser necesario, generar una alerta.

Los protocolos de la capa de transporte son: UDP y TCP.

UDP (user datagram protocol)	TCP (transmission control protocol)
<ul style="list-style-type: none"> <li>• Proporciona un servicio de entrega sin conexión y no confiable.</li> <li>• Utiliza IP para transportar los mensajes en la capa de red y agrega la capacidad de tener varios destinos dentro de una misma máquina.</li> <li>• La aplicación es la encargada de resolver la pérdida de paquetes, paquetes duplicados y la entrega desordenada de los mismos.</li> <li>• Se implementa este protocolo cuando se puede permitir la pérdida de paquetes (por ejemplo, en transmisiones de streaming).</li> </ul>	<ul style="list-style-type: none"> <li>• Es un servicio orientado a la conexión y es confiable.</li> <li>• Todos los paquetes de la conexión siguen el mismo camino y llegan en orden.</li> <li>• Cada paquete enviado requiere un reconocimiento (acknowledge) de que fue recibido correctamente.</li> <li>• Se suele emplear en situaciones donde no se permite la pérdida de paquetes.</li> </ul>

Tabla 1: comparación entre protocolos UDP y TCP

Fuente: Elaboración propia

La capa siguiente, la de aplicación, es la encargada de proporcionar servicios para los usuarios. Esta capa es relevante debido a que es la “responsable de proveer servicios de alta calidad para satisfacer las necesidades de los usuarios” (Cui. 2017).

Según Cui, los protocolos de la capa de aplicación definen la capacidad de mensajería y afectan el rendimiento del servicio. Se explican a continuación algunos protocolos que son relevantes para IoT:

- MQTT (Message Queueing Telemetry Transport)
- CoAP (Constrained Application Protocol)
- AMQP (Advanced Message Queuing Protocol)

**MQTT**

Es un protocolo de publicación-suscripción que facilita las comunicaciones de uno a muchos entre dispositivos a través de un "intermediario" MQTT central.

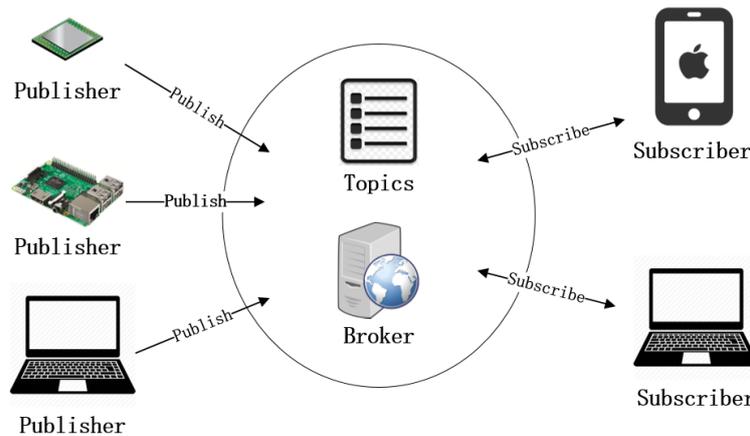


Figura 8: arquitectura de MQTT

Fuente: (Cui. 2017)

Este es el protocolo usado por AWS y se ha vuelto popular para IoT ya que es un protocolo liviano con un espacio de código pequeño adecuado para entornos restringidos (como, por ejemplo, dispositivos IoT integrados).

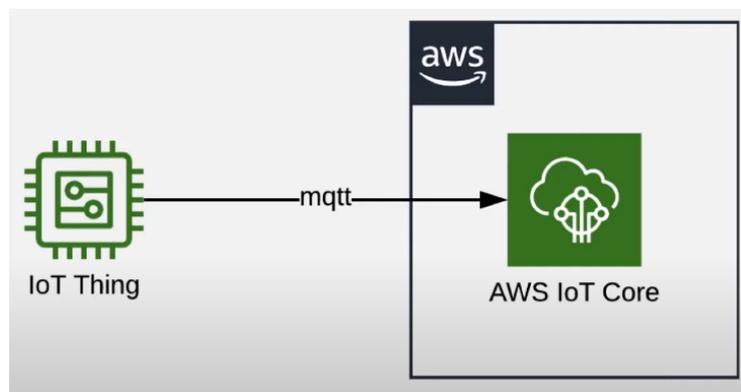


Figura 9: arquitectura IoT simple conectada a AWS

Fuente: («How to Get Started with AWS IoT Core Quick Connect». 2020)

AMQP

AMQP es un protocolo orientado a mensajes que sirve para pasar mensajes comerciales entre aplicaciones u organizaciones de manera confiable y segura.

Al igual que MQTT, AMQP es un protocolo de publicación-suscripción y, como se puede ver en la figura 10, se compone de: una cola (Queue), enlace (Binding) y un intercambiador (Exchange) que permiten enrutar los mensajes al receptor correcto.

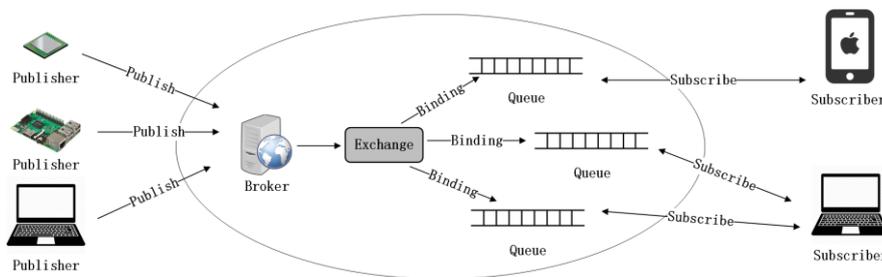


Figura 10: arquitectura de AMQP

Fuente: (Cui. 2017)

Uno de los protocolos que Azure IoT Hub permite usar para que los dispositivos IoT se puedan comunicar con su nube es el AMQP.

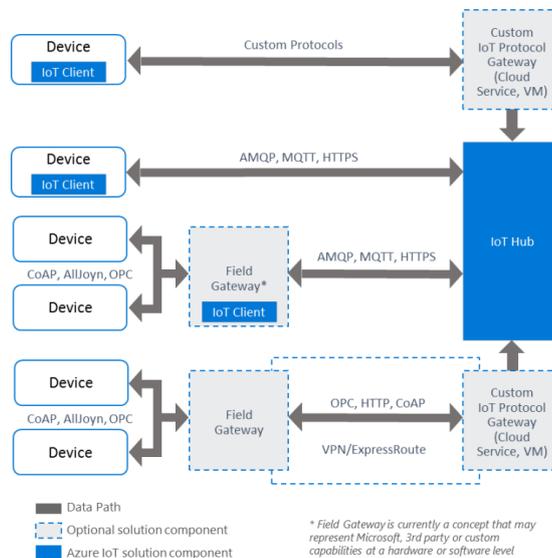


Figura 11: conectividad de dispositivos con Azure IoT Hub

Fuente: (Microsoft. 2021)

**CoAP**

CoAP es un protocolo IoT cliente-servidor sin estados basado en HTTP que permite a los clientes realizar solicitudes web, así como a los servidores responder a dichas solicitudes.

CoAP funciona gracias al uso de transacciones asíncronas (usando UDP) y a través de los métodos: POST, GET, PUT y DELETE.

Este protocolo trabaja con cuatro tipos de mensajes:

- Confirmable (CON) = son mensajes que se reenvían por tiempo de espera hasta que llegue la confirmación de envío exitoso, lo que garantiza la recepción de este.
- No confirmable (NON) = se envía y no se tiene garantía de entrega exitosa; tampoco hay reconocimiento de éxito.
- Reconocimiento (ACK) = confirman la finalización o el fracaso de un evento.
- Reinicio (RST) = son mensajes que están vacíos y son CON por naturaleza.

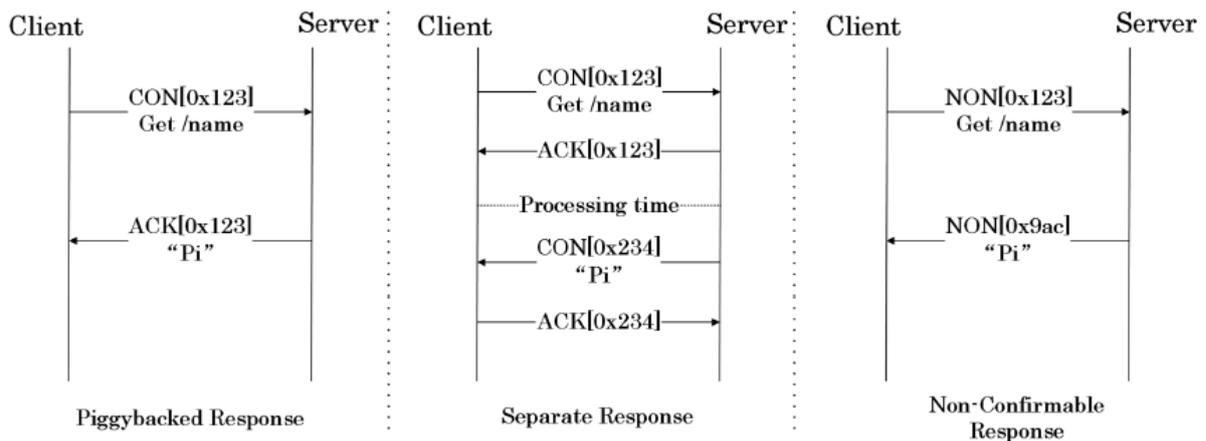


Figura 12: comunicación CoAP

Fuente: (Cui. 2017)

## 2.2. Estado del arte

### 2.2.1. Proyectos existentes para detectar incendios con ML

Con respecto a productos existentes en la Argentina relacionados con la problemática, está el proyecto “Zeroxygen - Track this fire” que propone un sistema de comunicación basado en redes de baja potencia y área amplia, integrando información libre de los satélites de NASA y sensores accesibles para generar avisos tempranos de incendios, así como seguimientos detallados del movimiento de fuego y humo (Schulkin. 2020)

Otra solución desarrollada en el país es “Satellites on Fire”; una aplicación que permite detectar incendios en tiempo real a través de inteligencia artificial e imágenes satelitales (Soler. 2021).

Lo que tienen en común ambas soluciones es el uso de imágenes satelitales para analizar las zonas y detectar el incendio.

Un ejemplo de proyectos ya existentes donde se combina IoT con machine learning para detectar incendios es el sistema “Firecatch” desarrollado en Chile. Dicho sistema busca detectar potenciales focos de incendio a través del uso de cámaras IP PTZ, sensores térmicos y sensores ópticos de alta resolución.

Estos dispositivos se conectan bajo una misma red y capturan información que es analizada en tiempo real por algoritmos de deep learning, disminuyendo así el tiempo de respuesta por parte de los entes responsables de tomar acción (ACOT System. 2021)

Como lo indican sus términos y condiciones, “Firecatch” está dirigido a usuarios residentes en Chile, y cumple con la legislación establecida en dicho país.

Por ende, lo que se busca con este PFI es enfocarse en un desarrollo dirigido netamente a la República Argentina, integrado dentro del INTEC a la línea de Investigación que explora cómo aplicar las soluciones de IoT y Machine Learning para el uso sustentable de los Recursos Naturales (Inchausti y otros. 2022) y, en particular, continúa el trabajo de la aplicación “AQUA” en dónde se desarrollaron Modelos de Machine Learning para la prevención de incendios forestales en el contexto del Partido de Pinamar (Martínez Saucedo. 2021), y que habilita la continuidad del trabajo para futuros investigadores de la UADE.

### 2.2.2. Algoritmos de reconocimiento de imágenes

Aparte de las CNN que se explicaron anteriormente, otros algoritmos de reconocimiento de imágenes son los Transformers.

A diferencia de las CNN, los Transformers son arquitecturas que requieren muchísimos más datos para poder ser entrenadas, pero al momento de predecir terminan siendo más eficientes como se puede ver en la figura 13. El paper “An image is worth 16x16 words: transformers for image recognition at scale” de Dosovitskiy y otros (2021) que se usó como base para este punto sirve para explicar cómo los Transformers utilizan como ingreso del dato un pequeño parche de la imagen (no la imagen completa) para el entrenamiento de la red.

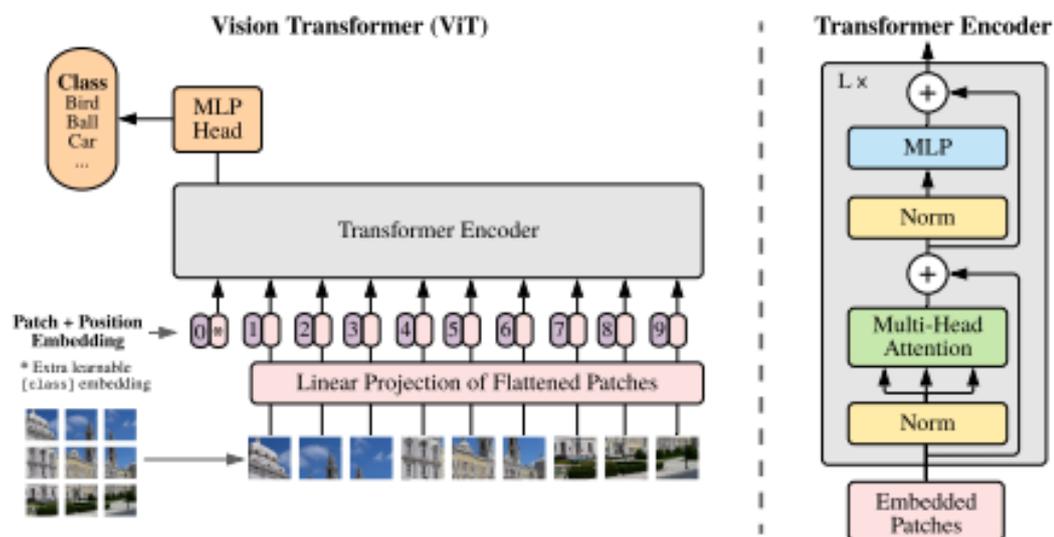


Figura 13: descripción general del modelo

Fuente: (Dosovitskiy y otros. 2021)

Al generar varias imágenes de una sola, como se ve en la figura 13, sección “Patch + Position Embedding” en la esquina inferior izquierda, se agranda significativamente la cantidad de datos para entrenar la red, lo que se traduce en que esta arquitectura supere en términos de eficiencia a las CNN (las más usadas hoy en día). No solo supera su performance, sino que también necesita significativamente menos tiempo de procesamiento como se observa en la figura 14.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	<b>88.55</b> ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet Real	<b>90.72</b> ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	<b>99.50</b> ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	—
CIFAR-100	<b>94.55</b> ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	—
Oxford-IIIT Pets	<b>97.56</b> ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	—
Oxford Flowers-102	99.68 ± 0.02	<b>99.74</b> ± 0.00	99.61 ± 0.02	99.63 ± 0.03	—
VTAB (19 tasks)	<b>77.63</b> ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	—
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Figura 14: comparación con el estado del arte en benchmarks de clasificación de imágenes

Fuente: (Dosovitskiy y otros. 2021)

## 2.3. User Research

### 2.3.1. Entrevistas

Para validar el alcance y la utilidad de la solución propuesta en este PFI, se decidió entrevistar a un experto en el tema.

Se le realizó una entrevista al Sr. Martin Quiroga, jefe de brigada y responsable del departamento de incendios forestales de la federación Mendocina desde hace 18 años. También trabaja como bombero voluntario desde hace 23 años para la federación de bomberos voluntarios.

El contacto con el Sr. Quiroga se realizó vía telefónica con una duración de 15 minutos. Martín pudo confirmar que hoy en día en la provincia de Mendoza los incendios forestales se detectan principalmente a través de 2 canales: uno es las llamadas telefónicas (provenientes de organismos oficiales o de transeúntes) y otro es el monitoreo manual usando páginas web de uso gratuito y público para detectar incendios, como FireMaps.

También se le comentó de la solución propuesta en el presente trabajo y su respuesta fue muy buena; le pareció que el enfoque era adecuado y que tener cámaras instaladas en puntos estratégicos es ideal porque sería equivalente a tener a un vigía monitoreando la zona. Ésta es una práctica que se mantiene vigente en los parques nacionales del país.

También se validó con él si el tiempo entre imágenes a ser evaluadas por el modelo era suficiente y su respuesta fue que las primeras 2 horas de un incendio son cruciales. Porque en

---

una hora puede pasar de ser un incendio que se puede controlar con una cuadrilla a volverse un verdadero caos.

Por último, se le preguntó sobre la proporción de falsas alarmas que reciben al mes y, si bien no sabía el porcentaje exacto, si pudo confirmar que es un número significativo.

La entrevista transcrita se puede encontrar en el anexo 11.2. de este PFI y ayudó a validar el alcance y enfoque de la solución propuesta de la mano de un experto en el área.

### **3. Descripción**

#### **3.1. Desarrollo de una red neuronal para detección de incendios a través de imágenes**

En esta etapa, se creó una red neuronal convolucional que es la encargada de recibir las imágenes, identificar patrones y en el proceso se ajustan los parámetros de la red hasta que aprenda a detectar si la imagen analizada contiene fuego o no.

Éste es un problema de clasificación (es decir, se busca predecir categorías), por lo que el tipo de aprendizaje que se aplicará será un aprendizaje supervisado, donde se utilizarán imágenes junto con su variable objetivo (fuego/no fuego) para el entrenamiento.

Para el desarrollo de la CNN, se utilizó la librería FastAI (fast ai. 2022), que es una librería de Deep Learning para creación de redes neuronales.

Una vez la CNN aprendió, se testea el rendimiento con una matriz de confusión para ver el porcentaje de falsos positivos y/o negativos.

##### **3.1.1. Set de datos a utilizar**

Para el entrenamiento de la red neuronal, se usaron datasets públicos de incendios forestales provenientes de plataformas libres como Kaggle.

Un ejemplo de los datos que fueron utilizados en este PFI es el dataset “Wildfire Detection Image Data”, que contiene imágenes de incendios forestales de libre uso para entrenar modelos de machine learning y deep learning (Dincer. 2021).

El dataset previamente mencionado se divide en 2 directorios:

- Datos de entrenamiento y validación: contiene 1.832 imágenes pre-etiquetadas (indicando si existe fuego o no) para poder entrenar y validar el modelo.
- Datos de testeo: contiene 68 imágenes para testear el modelo previamente entrenado. Con estos datos se espera analizar el rendimiento real del modelo.

Adicionalmente, se usaron imágenes de incendios reales en la Argentina para la etapa de testing. Así se pudo validar el rendimiento del modelo ante imágenes con la flora característica de los bosques en la Argentina.

También cabe resaltar que en el proceso de entrenamiento de la CNN se utilizaron imágenes de incendios forestales tomadas desde distintos ángulos para simular los 3 escenarios que se consideran en este PFI: cámaras fijas, puntos móviles y vista desde abajo, y se busca que la red tenga las capas intermedias necesarias para que pueda detectar patrones de humo y de fuego. Es decir, se espera poder detectar el fuego de manera temprana.



Figura 15: imágenes modelos de incendios forestales

Fuente: (Cruz y otros. 2016)

### 3.1.2. Preparación de los datos

El procedimiento de preparación de los datos consistió en cargar las 1.832 imágenes etiquetadas de Kaggle. Adicionalmente, para aumentar la cantidad de imágenes disponibles para el entrenamiento, se realizaron transformaciones a las imágenes.

Para realizar dichas transformaciones se utilizó la función “aug\_transforms” que sirve para crear una lista de transformaciones de: volteo, rotación, zoom, deformación e iluminación sobre el set de datos de entrenamiento.

En la figura 16 se puede ver algunos ejemplos de las transformaciones que se pueden obtener con “aug\_transforms” para aumentar la cantidad de datos disponibles y mejorar el aprendizaje del modelo, ya que no va a depender de propiedades cambiantes como: ángulo de la cámara, luminosidad, entre otros.

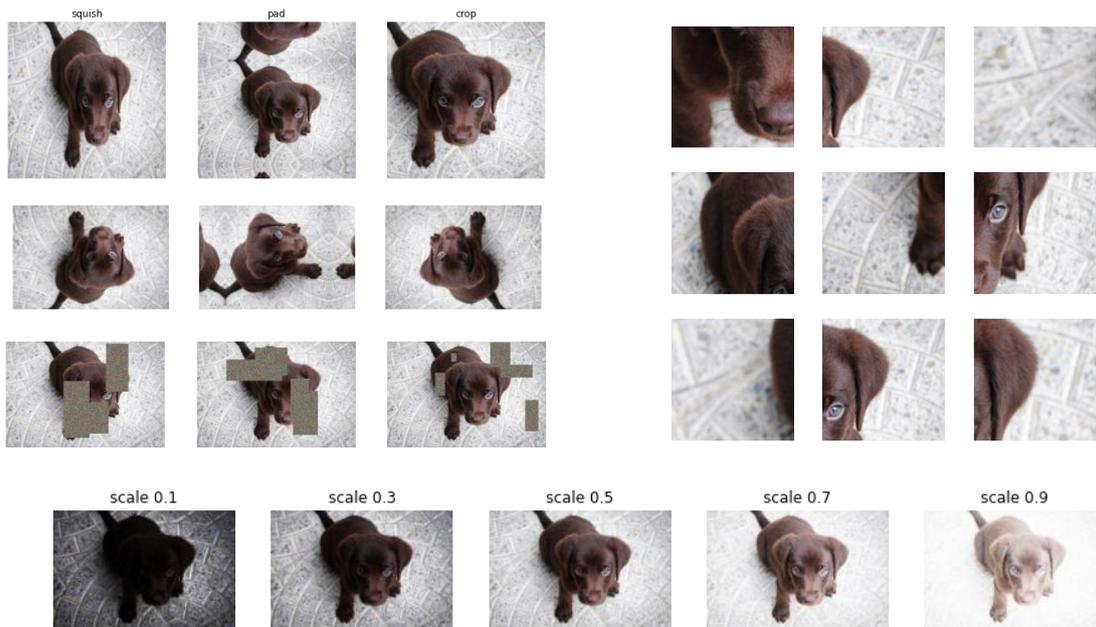


Figura 16: ejemplo de transformaciones usando “aug\_transforms” de FastAI

Fuente: (fast ai. 2022)

En la figura 17 se pueden ver algunas de las imágenes que serán utilizadas en la etapa de entrenamiento que ya pasaron por el proceso de preparación de datos:



Figura 17: imágenes listas para ser usadas en el entrenamiento de la red neuronal

Fuente: elaboración propia

### 3.1.3. Entrenamiento y evaluación del modelo

Una vez se prepararon los datos, se debe entrenar la red neuronal. En general, el proceso de entrenamiento consta de los siguientes pasos:

1. Se define la cantidad de capas y neuronas que conformarán la red
  - a. La primera capa (capa de entrada) debe tener la misma cantidad de neuronas que de valores de entrada tengo. En este caso serán los pixeles que tienen la imagen (250x250 pixeles).
  - b. La última capa (capa de salida) debe tener la misma cantidad de neuronas que de valores de salida. En este caso serán 2: “fuego” y “no fuego”
  - c. Para saber cuántas capas y conexiones intermedias (capas ocultas) se necesitan, hay que irlo testeando porque no hay forma de saber de antemano la cantidad de capas y conexiones óptima.
2. Se define la función de transferencia y la función de activación
  - a. En este caso, se probaron los algoritmos ResNet y ConvNeXt. Ambos algoritmos ya han sido pre entrenados por lo que no es necesario indicar la función de activación y de transferencia.
3. Se define el algoritmo que va a encontrar los pesos sinápticos
  - a. En este caso, se probaron los algoritmos ResNet y ConvNeXt.
  - b. Ambos algoritmos ya tienen ciertos pesos pre entrenados, por lo que no se empieza desde cero.

En el caso de la CNN desarrollada para este PFI, los parámetros que se modificaron fueron los siguientes:

- dls (data loader object): es donde están las imágenes para el entrenamiento.
- arch: es la arquitectura de la red que se va a aplicar. Para este PFI se utilizaron ResNet y ConvNeXt.
- Metric: esta es la métrica que se quiere usar para medir el rendimiento de la CNN. Para este caso se eligió el “error rate”.
- Learning rate: es un número que permite decidir cuanto se quiere que se ajusten los pesos sinápticos. Un learning rate pequeño implica que el factor delta con el que se van a ajustar los pesos en cada iteración es bajo (es decir, se irán a justando de a poco).

### 3.1.3.1. Proceso de selección del learning rate

El learning rate (o tasa de aprendizaje) es un número que permite indicarle al modelo cuanto se deben ajustar los pesos sinápticos después de cada iteración en la etapa de entrenamiento.

Un learning rate pequeño (ej.:  $10^{-7}$ ) implica que el factor delta con el que se van a ajustar los pesos en cada iteración es bajo (es decir, se irán ajustando de a poco).

Para seleccionar el learning rate de cada modelo entrenado y testeado en este PFI, se utilizó la función “lr\_finder()”. El learning rate finder (buscador de tasa de aprendizaje, en castellano) va a correr el modelo varias veces e irá aumentando el valor de learning rate en cada corrida. Luego graficará las pérdidas o losses (eje y) en función del learning rate (eje x) e indicará la tasa de aprendizaje recomendada.

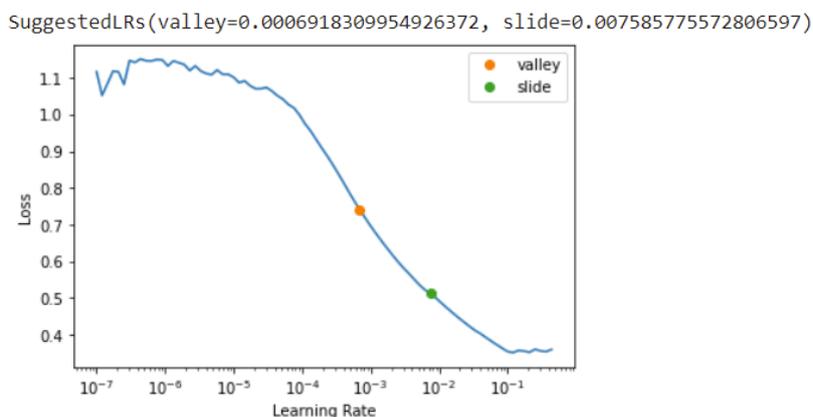


Figura 18: learning rate Finder obtenido en el entrenamiento de la red ConvNeXt

Fuente: elaboración propia

### 3.1.3.2. Entrenamiento y evaluación de la CNN

En esta sección se detallan las distintas versiones que se crearon y probaron hasta llegar a la CNN final que se usó en el producto final ‘StopFire’.

**Versión 1**

Para la primera versión, se creó una CNN con 34 capas intermedias. Los parámetros configurados fueron:

- dls: data loader con las imágenes precargadas
- arch: ResNet34
- Metric: error\_rate
- Learning rate: 0.0008317637839354575

En la matriz de confusión (figura 19) se puede ver que hubo 2 predicciones erróneas: un falso positivo y un falso negativo. Es decir, una imagen en la que no había fuego el modelo la clasificó como que si había fuego con un 93% de probabilidad; y otra imagen en la que había fuego, el modelo la clasificó como “sin fuego”, con un 68% de probabilidad (figura 20).

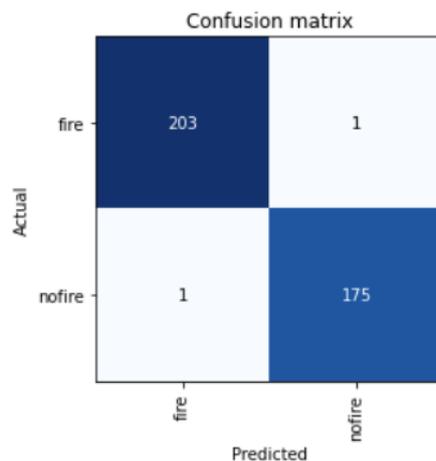


Figura 19: matriz de confusión del modelo ResNet34

Fuente: elaboración propia



Figura 20: falsos positivos y negativos

Fuente: elaboración propia

Como se observa en la figura 20, el falso positivo es consecuencia de la neblina que se observa en el bosque: se puede inferir que la CNN confundió neblina con humo. Mientras que con el falso negativo se puede observar lo contrario: la CNN confundió humo con neblina.

Dada la naturaleza de la problemática de este PFI, no se puede aceptar que el modelo genere falsos negativos, porque si las imágenes se van a analizar cada 30 minutos y llegase a confundir humo con neblina, no será hasta la segunda captura y envío de foto en los siguientes 30 minutos que se va a generar la alerta en el dashboard, lo que implicaría que el fuego ya lleva 1 hora de haberse iniciado y puede escalar muy rápidamente en un desastre difícil de contener.

## Versión 2

En la segunda prueba, se agregaron 16 capas intermedias más, para un total de 50. Los parámetros configurados fueron:

- dls: data loader con las imágenes precargadas
- arch: ResNet50
- Metric: error\_rate
- Learning rate: 0.0004786300996784121

Al igual que con la versión 1, se ve como la matriz de confusión sigue arrojando predicciones erróneas; esta vez 5 falsos positivos y 2 falsos negativos (ver figura 21)

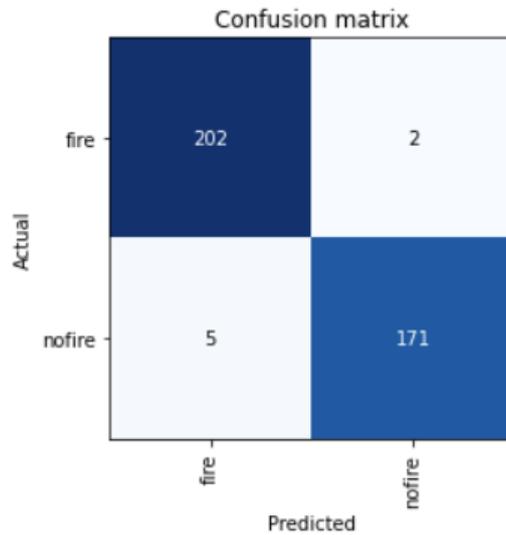


Figura 21: matriz de confusión del modelo ResNet50

Fuente: elaboración propia



Figura 22: falsos positivos y negativos

Fuente: elaboración propia

Al igual que con la versión 1, el modelo tuvo dificultades para diferenciar neblina de humo. Adicionalmente, se puede observar en la figura 22 como también tuvo problemas aprendiendo a detectar patrones de atardeceres.

De nuevo, no se puede aceptar que el modelo genere falsos negativos, por lo que se hicieron más pruebas para generar una tercera versión.

**Versión 3**

Como tercera versión, se duplicaron la cantidad de capas intermedias y se mantuvo el Learning rate anterior. Los parámetros configurados fueron:

- dls: data loader con las imágenes precargadas
- arch: ResNet101
- Metric: error\_rate
- Learning rate: 0.0010000000474974513

La matriz de confusión indica que esta vez no hubo ningún falso positivo ni falso negativo, lo que es una buena señal de como la red logró aprender a detectar mejor los patrones de hubo, fuego, neblina, atardeceres, etc. (ver figura 23).

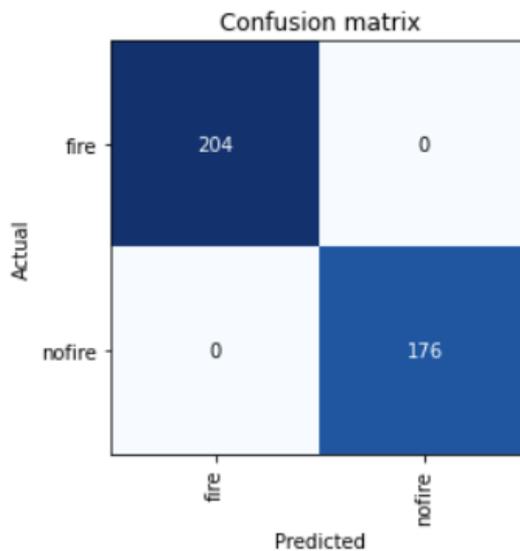


Figura 23: matriz de confusión del modelo ResNet101

Fuente: elaboración propia

Una vez que se logró llegar a una CNN que aparenta ser fiable, se aplicó la función “to\_fp16()”. Esto lo que hace es transformar todo el modelo en formato de coma flotante de precisión media; es decir, los pesos sinápticos, las entradas y cualquier número flotante lo almacena en el tipo de dato float16 en vez de float32.

Lo que se busca con esto es reducir el uso de memoria y reducir los tiempos de entrenamiento del modelo. La desventaja es que, como su nombre lo indica, se disminuye la precisión a la mitad.

Una vez transformada la CNN ResNet101 a float16, se volvió a testear y, como se puede ver en la matriz de confusión, arrojó 2 falsos positivos y ningún falso negativo.

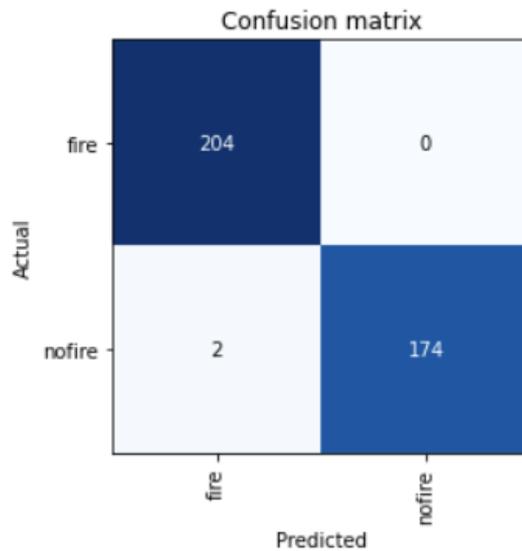


Figura 24: matriz de confusión del modelo ResNet101 to\_fp16

Fuente: elaboración propia



Figura 25: falsos negativos

Fuente: elaboración propia

Al disminuir la precisión a la mitad, se repite lo que sucedió en la versión 2 en donde la red no pudo aprender a diferenciar el atardecer del fuego (esta vez con una probabilidad del 93% y 82% para cada caso).

Por otro lado, una ventaja de este modelo es que no hubo ningún falso negativo pese a la reducción de la cantidad de decimales de los pesos sinápticos. Lo que quiere decir que pudo detectar patrones con un formato de precisión media (float16).

#### Versión 4

Como última versión, se creó una red ConvNeXt en vez de ResNet. Los parámetros configurados fueron:

- dls: data loader con las imágenes precargadas
- arch: convnext\_small\_in22k
- Metric: error\_rate
- Learning rate: 0.007585775572806597
- Se transformó a float16 para reducir el uso en memoria y tiempo de entrenamiento.

ConvNeXt, a diferencia de ResNet, no va a forzar la conversión de la imagen original a una de 250x250 pixeles, sino lo que hace es “cortar” una parte de la imagen de tamaño 250x250. De esa forma, se aprovecha mejor la imagen original sin necesidad de hacerle transformaciones adicionales.

Otra ventaja de ConvNeXt es que las transformaciones usando “aug\_transforms” las generará en cada epoch. De esta forma se puede asegurar de no ver la misma imagen varias veces, así como disminuir la probabilidad de sobre aprendizaje (también conocido como overfitting).

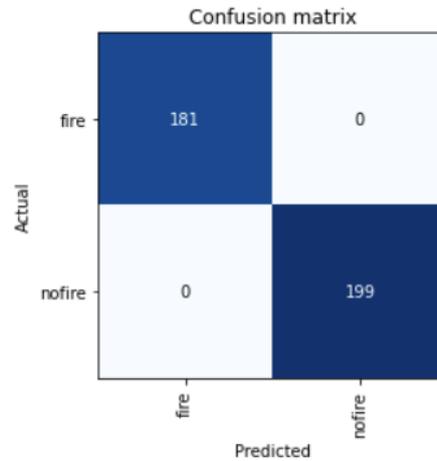


Figura 26: matriz de confusión del modelo ConvNeXt

Fuente: elaboración propia

### 3.1.4. Conclusiones

Después de generar distintos modelos y comparar los resultados con la matriz de confusión, se eligió la versión 4: CNN con ConvNeXt.

Se consideró muy importante que la red no arrojara falsos negativos, es decir, que marcara como “no fire” imágenes en las que, si había fuego o humo, porque esto implicaría que al momento en que empieza el fuego y la cámara realice el envío de la imagen para ser analizada en vivo, no se detectaría el incendio y tendrían que pasar 30 minutos más para que se mande la alerta. El accuracy obtenido con ConvNeXt fue del 100%, por lo que se decidió elegir ese modelo como el definitivo.

### 3.2. Dashboard de alerta de incendios

Para centralizar y accionar la información capturada por las cámaras que conforman la red de monitoreo, se desarrolló un dashboard web para ser utilizado por los bomberos y entes responsables de tomar acciones a la hora de detectar un incendio forestal.

### 3.2.1. Mockups

Se diseñaron bosquejos para el dashboard de alerta y monitoreo de incendios a ser utilizado por los bomberos y personal responsable del monitoreo de la zona.

#### Mapa de cámaras

Esta sección muestra el mapa con la zona que se está monitoreando, la ubicación (markers) de las cámaras IP en un mapa, junto con un resumen de cuantos incendios fueron alertados en las últimas 24 horas y la cantidad de cámaras instaladas en la red.

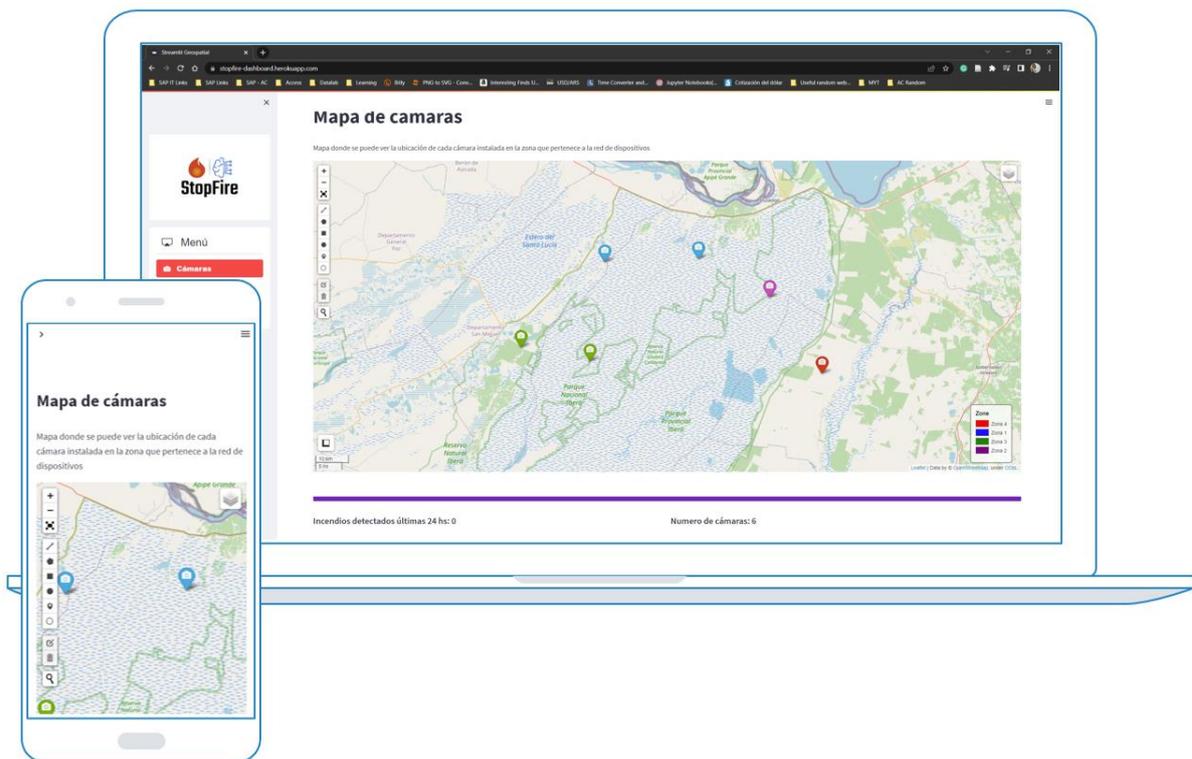


Figura 27: página principal – Mapa de cámaras

Fuente: elaboración propia

Módulo de incendios

Esta sección permite a los usuarios filtrar por rango de fechas, ubicación donde se detectó el incendio y tipo de cámara (si es un punto fijo o móvil).

Una vez se elija los filtros, se actualiza el mapa y se muestran las alertas que cumplan con los filtros preseleccionados.

Además, se le permite al usuario ver la foto que el modelo clasificó como “incendio” e indicar si la imagen fue categorizada incorrectamente (es decir, si era un falso positivo). De esta forma se podrán utilizar esos falsos positivos en entrenamientos de versiones futuras del modelo.

Por último, se podrá ver en un mapa la ubicación geográfica de los incendios y la distribución con colores por mes de ocurrencia.

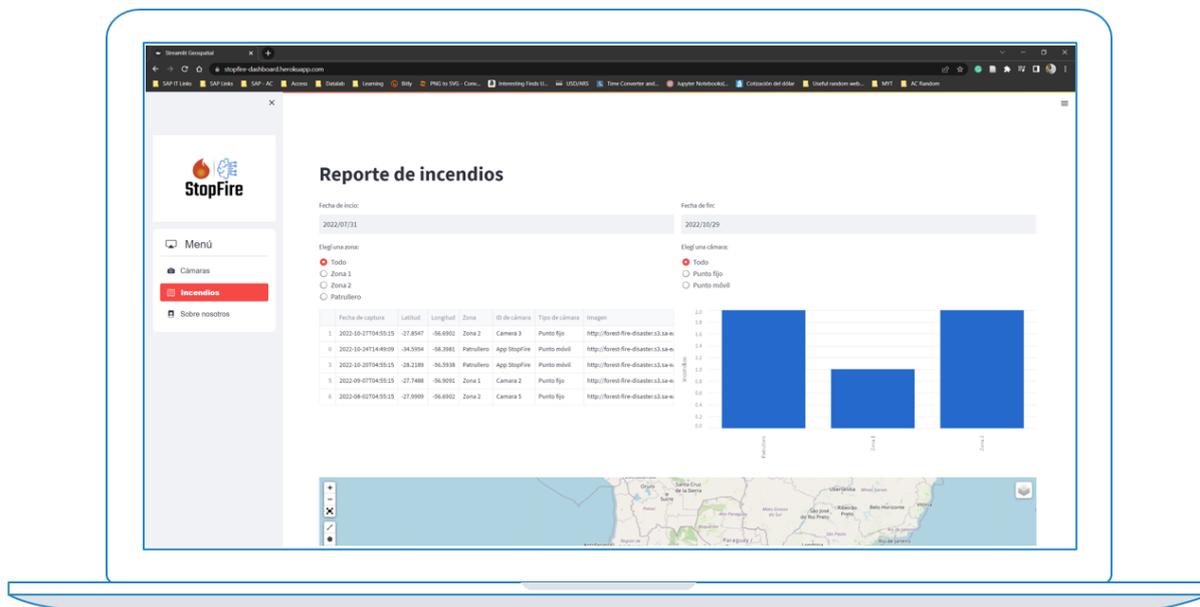


Figura 28: módulo de incendios – Reporte de incendios

Fuente: elaboración propia

Página institucional “Sobre nosotros”

Esta sección ofrece una breve descripción del equipo que desarrolló la aplicación StopFire.

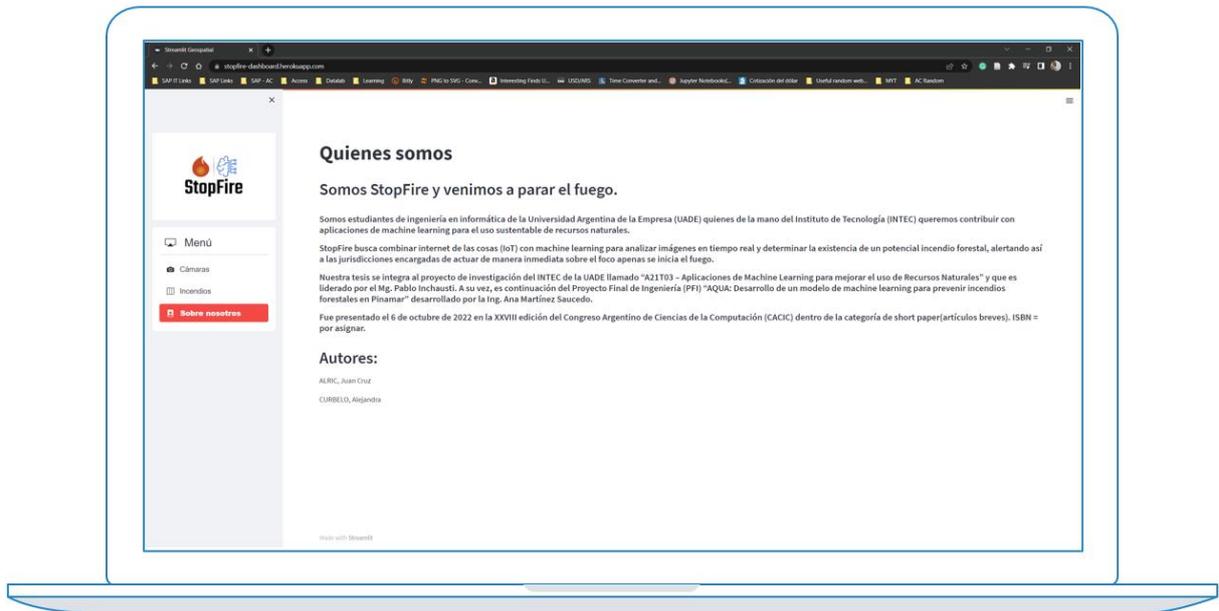


Figura 29: página institucional con información del proyecto

Fuente: elaboración propia

**3.3. Aplicación móvil**

Se desarrolló una aplicación móvil para que los patrulleros voluntarios puedan informar de incendios que estén sucediendo cerca su zona, adjuntando una foto en vivo.

La aplicación envía al servidor la imagen, la geolocalización del dispositivo y la fecha y hora de la captura de la foto para alertar a las autoridades.

El objetivo de esta aplicación es ayudar a los bomberos y entes responsables a identificar incendios que ocurran en zonas fuera de las monitoreadas por las cámaras que pertenecen a la red de dispositivos, ampliando así el alcance de la red.

### 3.3.1. Wireframes

Se diseñaron los esquemas de alta fidelidad para la aplicación móvil que sirven para mostrar de manera grafica el flujo que va a seguir el usuario de la aplicación (patrullero voluntario).

En la figura 31 se puede observar la pantalla de inicio que aparece una vez se accede a la aplicación (fig.30). Contiene un botón en el medio para que el usuario pueda iniciar la cámara y generar una alerta de incendio.



Figura 30: wireframes. Landing page

Fuente: elaboración propia

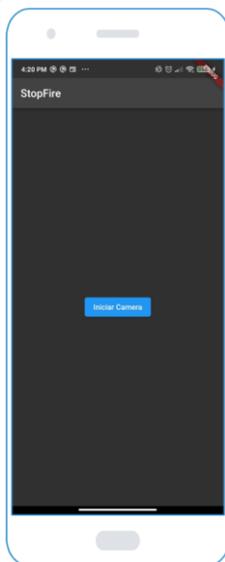


Figura 31: wireframes. Pantalla de inicio

Fuente: elaboración propia

Una vez se selecciona el botón “iniciar cámara”, se va a activar la cámara junto a los botones “Volver” y “Enviar”, dándole al usuario la opción de regresar a la pantalla anterior o de capturar la imagen y enviarla (ver figura 32).



Figura 32: wireframes. Capturar foto de incendio

Fuente: elaboración propia

Al seleccionar “Enviar”, aparece un mensaje que indica que la imagen fue enviada exitosamente.

El usuario puede regresar a la página de inicio para poder realizar otra alerta, en caso de que sea necesario, seleccionando el botón “Volver” (ver figura 33).



Figura 33: wireframes. Mensaje de confirmación de alerta

Fuente: elaboración propia

Por último, en caso de que haya un error con el servidor o con la aplicación, se va a mostrar un mensaje indicando que la alerta no se pudo realizar correctamente y que el usuario debe intentarlo de nuevo (ver figura 34).



Figura 34: wireframes. Mensaje de error

Fuente: elaboración propia

### 3.4. Arquitectura

Para este PFI se utilizó la nube de Amazon AWS porque provee el conjunto de servicios especializados en IoT más completo del mercado. Adicionalmente, ya se cuentan con conocimientos en la plataforma, lo que disminuye el tiempo invertido en aprendizaje por parte de los autores de este trabajo.

#### 3.4.1. Diagrama de arquitectura

La arquitectura propuesta para StopFire se puede observar en la figura 35. Se desarrolló el front end de la página web usando Streamlit y el de la aplicación móvil en Flutter porque ya se contaba en el equipo con conocimiento en ambas herramientas.

Para el back end se utilizó Flask puesto que facilita el desarrollo de aplicaciones web bajo el patrón MVC (Modelo Vista Controlador). Flask es el encargado de almacenar las imágenes categorizadas como ‘incendio’ por el modelo en un bucket de Amazon S3 para poder ser visualizadas por el usuario al momento de que se genere una alerta.

Para guardar toda la información de los incendios y de las cámaras de la red se utilizó una base de datos relacional en MySQL. El pilar de machine learning fue desarrollado en Python, usando la librería Fast.ai.

Por último, se utilizó la API de OpeStreetMap desde Streamlit para poder mostrar en los módulos de ‘Cámaras’ e ‘Incendios’ los mapas señalando la ubicación de las cámaras y las zonas donde se detectó el potencial incendio.

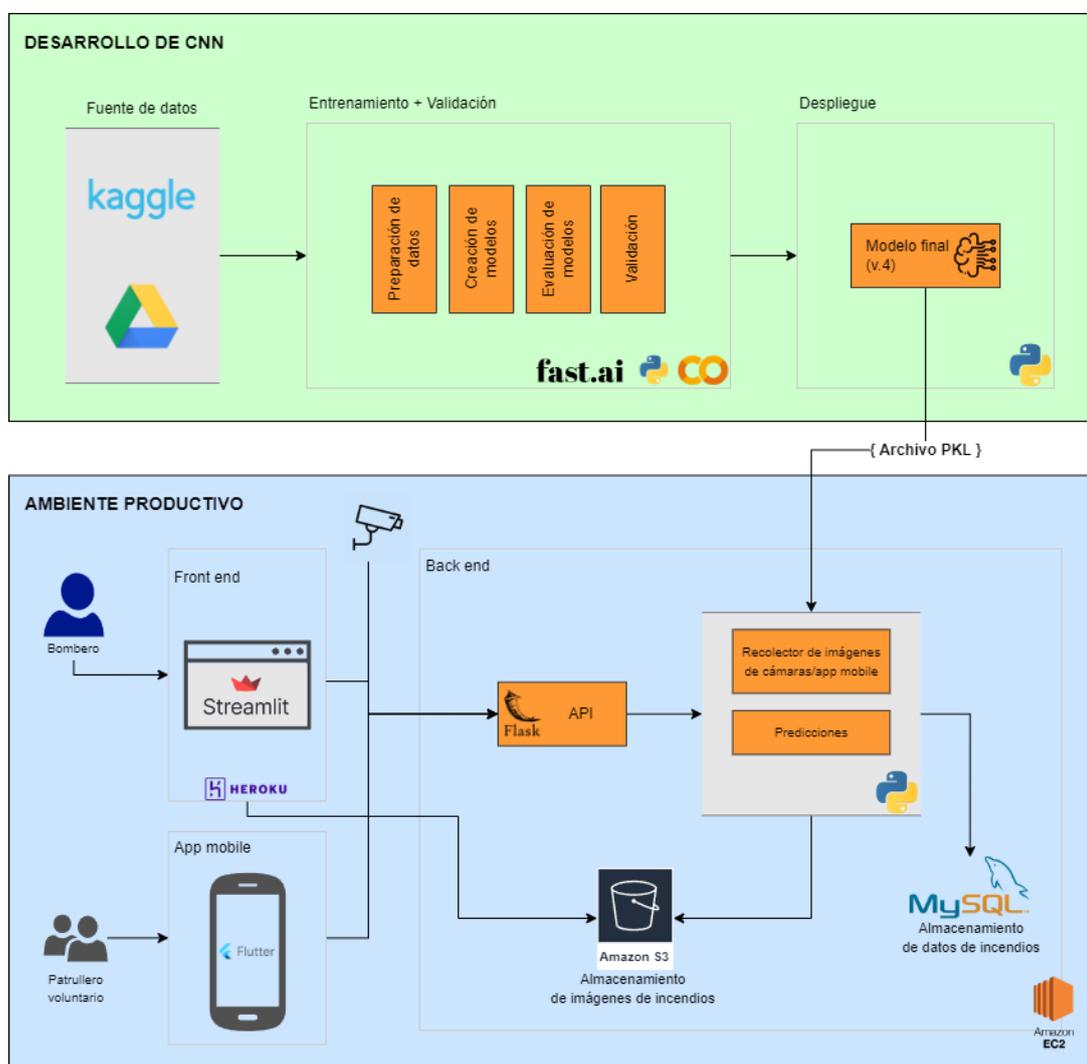


Figura 35: arquitectura de StopFire

### 3.4.2. Diagrama de base de datos

Se creó la base de datos en MySQL para almacenar la información de las cámaras junto con la información de las imágenes que la CNN clasificó como ‘incendio’.

Adicionalmente, se almacenará la foto real ya clasificada en un bucket (contenedor) de Amazon S3 y el campo “image” de la tabla “images” va a guardar la dirección web con la ubicación del archivo .jpg. De esa forma, cuando el usuario seleccione “Ver foto” desde el módulo de incendios, se podrá acceder a la fotografía original que la red neuronal utilizó para determinar la existencia de fuego y/o columnas de humo.

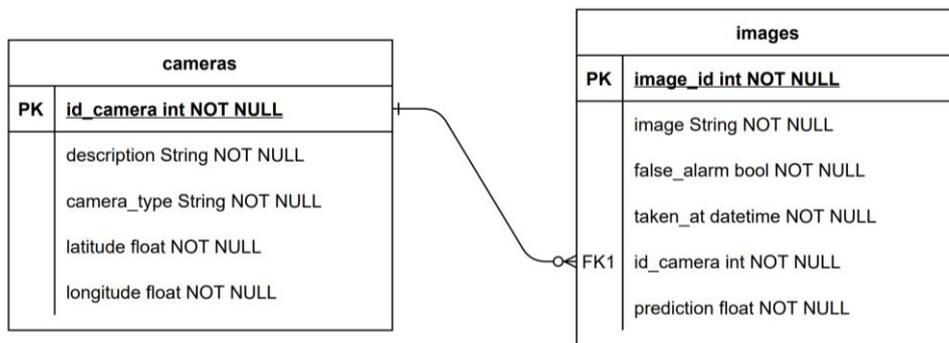


Figura 36: diagrama de base de datos de StopFire

Fuente: elaboración propia

### 3.5. Dispositivos IoT para la detección temprana de incendios forestales

La propensión “extrema” de incendios en el país, de acuerdo con el Servicio Meteorológico Nacional (SMN), abarca gran parte del territorio argentino (como se aprecia en la figura 37), por lo que se convierte en una preocupación para los habitantes de estas regiones.

Los principales rubros afectados por incendios forestales son: la agricultura, ganadería, turismo y cría de animales, ya que son actividades desarrolladas en terrenos poco vigilados y que poseen un alto índice de propagación de fuego. Es por ello por lo que se realiza en este PFI un análisis de aplicación de dispositivos IoT enfocado en la detección temprana de incendios.

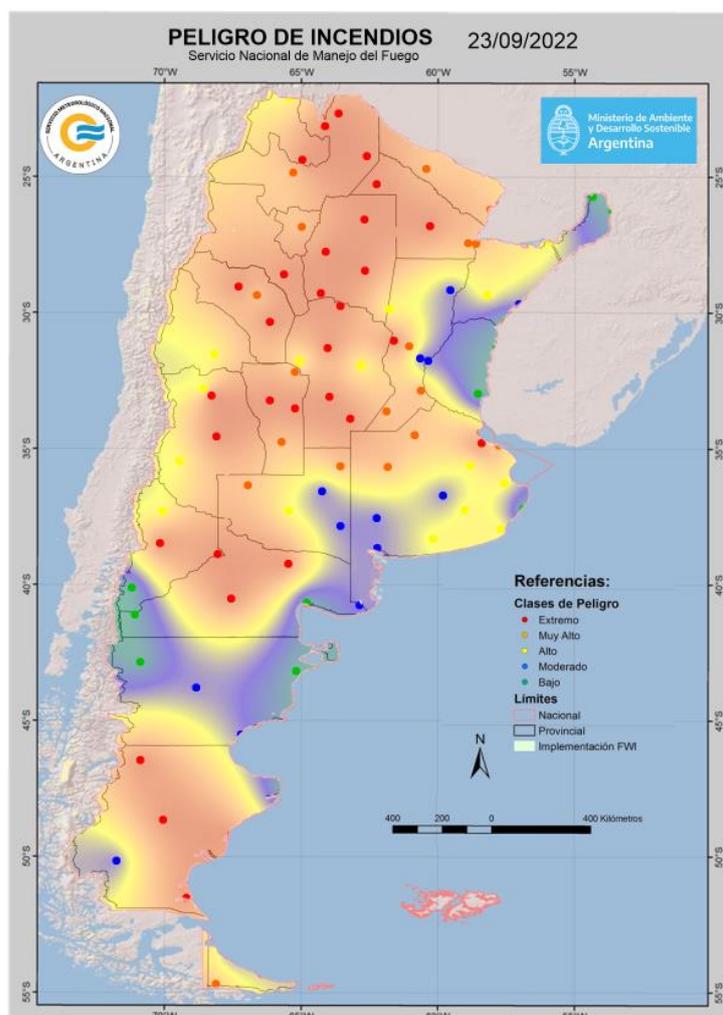


Figura 37: mapa de índices de peligro de incendios en Argentina

Fuente: (Servicio Meteorológico Nacional. 2022)

El uso de dispositivos IoT (cámaras y/o sensores) que deben ser posicionados a nivel de terreno, requieren tener un alcance visual hacia las zonas que se quiere vigilar, por lo cual se deben instalar en el punto más alto posible y acorde al terreno, con el fin de mantener una línea de vista limpia y cumplir su objetivo. En Argentina, las zonas de alta propensión de incendio coinciden en su mayoría con zona considerablemente planas, por lo que se considera viable la aplicación de un modelo de análisis con dispositivos IoT para la detección de incendios

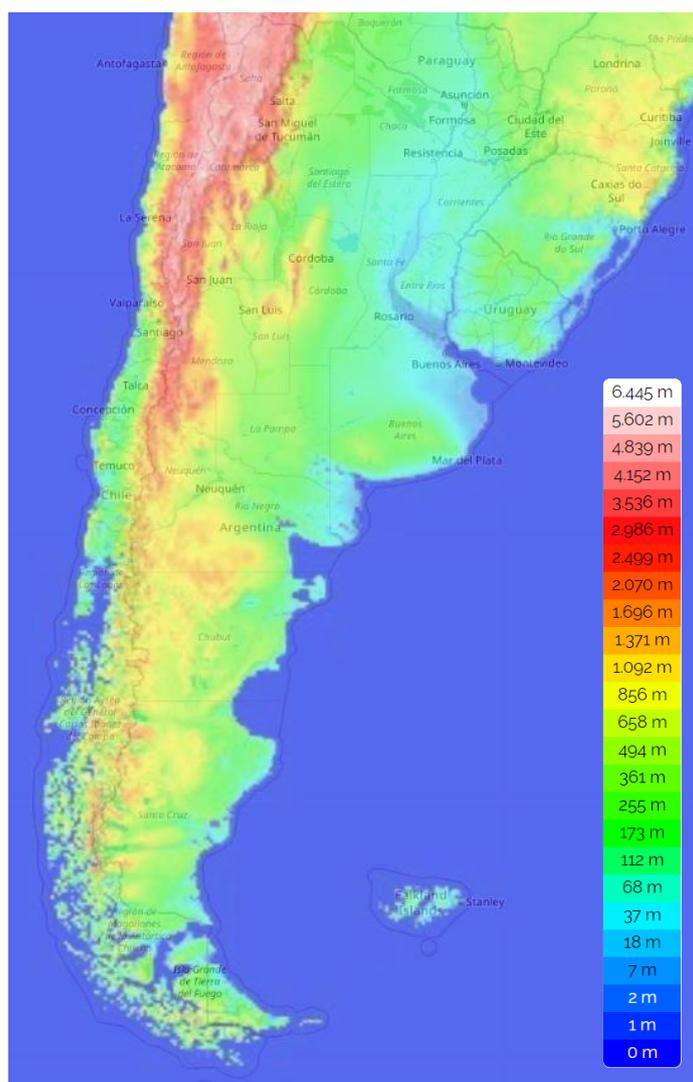


Figura 38: mapa topográfico de Argentina

Fuente: (topographic-map. 2022)

### 3.5.1. Dispositivos existentes

#### 3.5.1.1. Definición y características

Los dispositivos que se pueden utilizar en la detección temprana de incendios forestales se dividen en tres: satélites, cámaras, y sensores.

#### **Satélites**

Para capturar y analizar información casi en tiempo real (1-3 horas) de incendios forestales, la NASA pone a disposición para uso libre las imágenes y datos capturados por los satélites que alimentan el sistema FIRMS ('Fire Information for Resource Management System', en inglés).

FIRMS fue desarrollado en el 2007 por la Universidad de Maryland con fondos del Programa de Ciencias Aplicadas de la NASA y la Organización de las Naciones Unidas para la Agricultura y la Alimentación y fue transferido a la NASA en el 2012 (Earth Science Data Systems. 2021).

FIRMS cuenta con información proveniente de cuatro satélites (Aqua, Terra, S-NPP, y NOAA 20) que utilizan observaciones capturadas con los instrumentos MODIS (por las siglas en inglés de 'Espectro radiómetro de imágenes de resolución moderada') y VIIRS (por las siglas en inglés de 'Conjunto de radiómetros de imágenes infrarrojas visibles') para detectar incendios activos y anomalías térmicas en tiempo real. Esta información está disponible a nivel mundial dentro de las 3 horas posteriores a la observación. Con excepción de Canadá y Estados Unidos, donde la información está disponible en tiempo real (es decir, en menos de 1 hora) (Earth Science Data Systems. 2017).



Figura 39: flota de observación de la Tierra de la NASA donde se aprecian los 4 satélites que alimentan FIRMS

Fuente: (Platnick. 2017)

En la figura 40 se puede observar un ejemplo de una imagen satelital procesada y analizada por FIRMS donde se observan múltiples puntos de calor detectados por MODIS el 20 de agosto del 2020.



Figura 40: puntos de calor detectados

Fuente: (Earth Science Data Systems. 2017)

El primero de marzo del 2022, la Oficina Nacional de Administración Oceánica y Atmosférica de los Estados Unidos puso en órbita un nuevo satélite meteorológico llamado GOES-T (renombrado GOES-18) que servirá para rastrear fenómenos meteorológicos en la zona oeste de Estados Unidos, Alaska, Hawái, México, Centroamérica y el océano Pacífico (Strickland. 2022). Se espera que las primeras imágenes y datos capturados por GOES-18 estén disponibles en agosto de 2023.

GOES-18 podrá localizar puntos de calor para identificar incendios de manera temprana, predecir el movimiento del fuego, estimar la intensidad de un incendio y monitorear la producción de humo y los efectos del humo en la calidad del aire (Goddard Media Studios. 2022).



Figura 41: representación artística de una nave espacial GOES en órbita alrededor de la Tierra

Fuente: (Strickland. 2022)

Algunos ejemplos de proyectos desarrollados en la Argentina donde se utilizaron los satélites de FIRMS para la detección de incendios forestales son los citados anteriormente: “Zeroxygen - Track this fire” y “Satellites on Fire”.

En Zeroxygen se integra información capturada por satélites con datos de sensores para generar avisos tempranos de incendios, así como seguimiento detallado del movimiento de fuego y humo (Schulkin. 2020), mientras que Satellites on Fire combina inteligencia artificial con imágenes satelitales, brindando información de múltiples barridos satelitales y mejorando así el monitoreo del fuego (Soler. 2021).

La principal ventaja del uso de estos dispositivos para detectar fuego en la Argentina es que no se requiere una inversión monetaria en hardware, sino solamente en el desarrollo del software que va a traducir los datos capturados por los satélites en información accionable por la correspondiente jurisdicción.

Por otro lado, la desventaja del uso de satélites es que, en Argentina, la actualización de los puntos de calor detectados por FIRMS están disponibles en el sistema web FireMaps en un plazo de 1 a 3 horas posteriores a la observación. Adicionalmente, el área del incendio debe superar un kilómetro cuadrado para poder ser detectado, lo cual repercute en la propagación del incendio y en el margen de acción de los bomberos y rescatistas.

## **Cámaras**

Para detectar incendios con cámaras se pueden utilizar cámaras termográficas. Estas cámaras utilizan sensores infrarrojos para medir la energía infrarroja que emite un objeto cualquiera. Luego, transforman los datos capturados por el sensor en imágenes que muestran la temperatura aparente de la superficie de dicho objeto.



Figura 42: ejemplo de cámara termográfica

Fuente: (Promax. 2019)

Algunos proveedores nacionales de este tipo de cámaras son Hikvision y Bosch. Si bien las cámaras que ofrecen estas compañías pueden detectar fuego y humo, no son dispositivos especializados para la detección de incendios en ambientes abiertos, por lo que no son adecuadas para el campo de estudio de este PFI. Además de que tienen un costo 9 veces mayor que el de una cámara IP común.

Otro tipo de cámaras que se puede utilizar para detectar incendios forestales son las cámaras inteligentes como AWS DeepLens, que es una cámara especializada en aprendizaje profundo que permite ejecutar modelos de deep learning de manera local en el dispositivo para tomar acciones en función de lo que se captura.

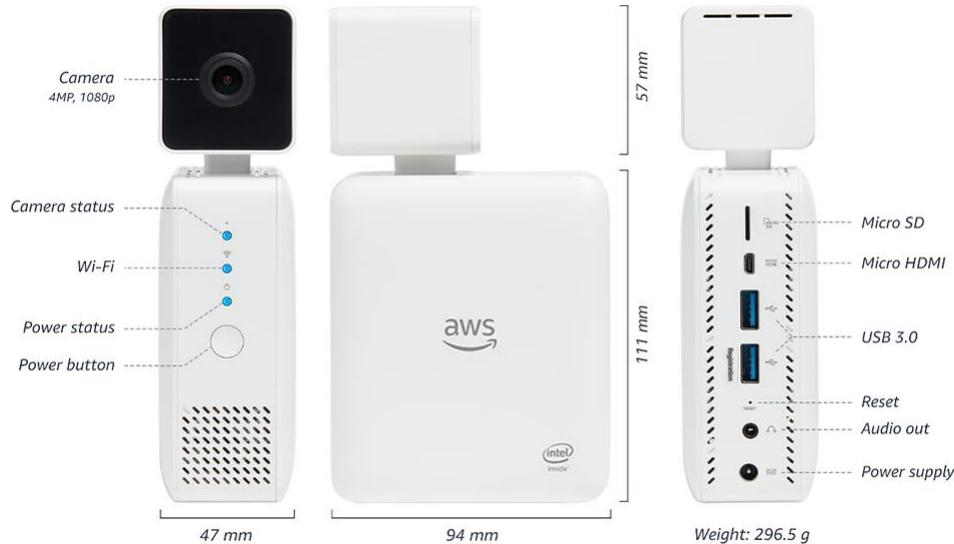


Figura 43: AWS DeepLens

Fuente: (AWS. 2019)

El uso de cámaras para la detección temprana de incendios forestales se ha vuelto una práctica común en países como Estados Unidos y Australia.

En el 2021, la startup con sede en San Francisco, Pano AI, está instalando cámaras panorámicas en las cimas de las montañas de California para ayudar a detectar señales de incendios forestales. El sistema utiliza cámaras giratorias 360 para tomar 10 fotos del paisaje cada 60 segundos, unirlos de manera secuencial y analizarlas para detectar columnas de humo mediante el uso de inteligencia artificial (Thomas. 2021).

Como prueba piloto, Pano AI instaló 23 cámaras en cuatro estados del territorio estadounidense (California, Colorado, Oregón y Montana).



Figura 44: instalación de la estación Pano en Healdsburg, California

Fuente: (Pano AI. 2022)



Figura 45: detección de columnas de humo usando el sistema Pano

Fuente: (Pano AI. 2022)

## Sensores

El uso de sensores para detectar incendios forestales de manera temprana tiene como objetivo monitorear variables que indican la existencia del fuego. Esas variables pueden ser: químicas, físicas o meteorológicas.

Entre las variables físicas se encuentran la emisión infrarroja y el calor. Las variables químicas son: CO<sub>2</sub>, CO, y emisiones de humo; y las meteorológicas: viento, presión atmosférica, temperatura, y humedad relativa (Hernández-Hostaller. 2017).

Como lo indica Hernández-Hostaller (2017), se pueden utilizar sensores para la detección de partículas de humo en el aire (variable química) aplicando el método óptico donde se utiliza un láser y un receptor de luz. Ante la presencia del humo, la luz llegará al receptor opaca y con menor intensidad, disparándose así una alarma.



Figura 46: detección fotoeléctrica de humo

Fuente: (Hernández-Hostaller. 2017)

También, se pueden utilizar de sensores para detectar emisiones infrarrojas y térmicas (variables físicas). Estos sensores tienen una alta sensibilidad lo que permite la detección a grandes distancias (se pueden usar de manera aérea, desde torres de observación o desde satélites en el espacio). Hay que tener en cuenta que cuando se utilizan en tierra, estos sensores disminuyen su efectividad ya que la vegetación y el follaje en la zona actúan como obstáculos que atenúan la emisión infrarroja, por lo que no es recomendable instalarlos en el terreno que se desee monitorear (Hernández-Hostaller. 2017)

Las ventajas del uso de sensores sobre cámaras o satélites radican principalmente en el costo, ya que suelen ser dispositivos más económicos. Requieren de un ancho de banda inferior y su consumo eléctrico es significativamente menor.

Un ejemplo de uso de sensores para detectar incendios forestales es el sistema desplegado por ARANTEC (casa de software española especializada en redes y sensores inalámbricos) en Extremadura, España.

Este sistema de detección utiliza sensores para captar información del entorno para predecir el comportamiento del fuego. La solución consta de (ARANTEC. 2021):

- Una estación meteorológica con sensores de temperatura, humedad relativa, presión atmosférica, viento, precipitación, radiación, cámaras de visión diurna y nocturna, data logger y panel solar, además de los elementos de anclaje y protección necesarios.
- Una red de sensores distribuidos en 10 puntos y localizados sobre el suelo, mástiles o los propios árboles. Está integrada por sensores de temperatura y humedad del suelo, humedad foliar y gases (CO y CO<sub>2</sub>), junto con los protocolos de comunicación que habilitan la transmisión de la información (4G, LoRaWAN, etc.). El sistema está configurado para enviar la información cada 10 minutos.



Figura 47: solución de ARANTEC para la detección de incendios forestales en Extremadura

Fuente: (ARANTEC. 2021)

Como se observa en la figura 48, dependiendo del tipo de variable que se desee analizar, se pueden aplicar distintos métodos de monitoreo. Cuando se combinan múltiples métodos de captura de información, se puede hacer frente al incendio de manera más eficaz que si se dependiera solamente de una fuente de información (por ejemplo, usar exclusivamente datos satelitales).

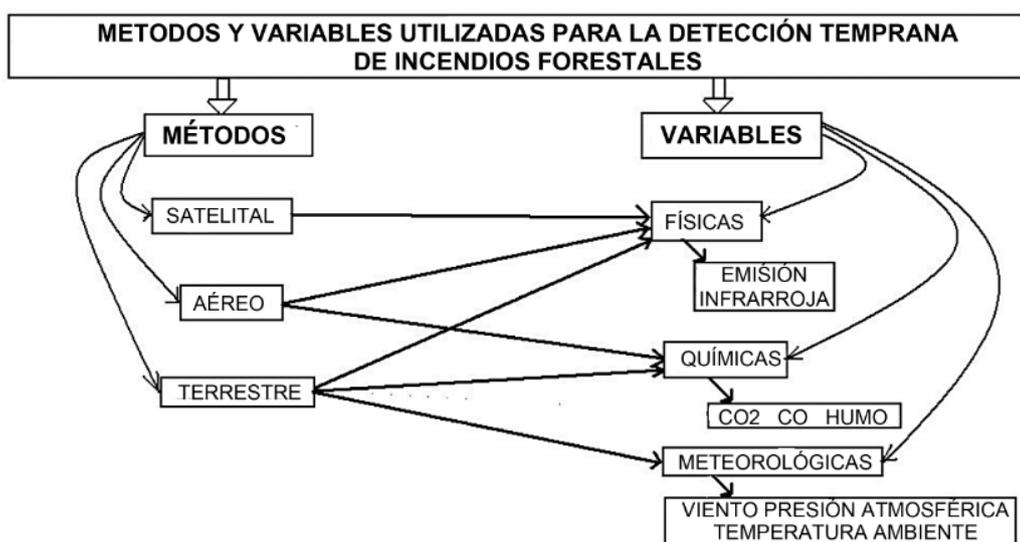


Figura 48: métodos y variables utilizadas para la monitorización del fuego en el bosque

Fuente: (Hernández-Hostaller. 2017)

Como ejemplo de una solución que utiliza distintos métodos de detección para combatir incendios forestales, se encuentra la que está siendo desarrollada por el gobierno de Nueva Gales del Sur (NSW), quienes asignaron en el 2021 30 millones de dólares en investigación y desarrollo de tecnología de incendios forestales para los próximos 4 años (Rabe. 2021). El objetivo es crear robots que puedan asistir a bomberos y socorristas en la zona, así como drones que ayuden a detectar rutas de escape para facilitar la evacuación de personal en el área en peligro y, por último, el uso de satélites que sirvan para predecir el movimiento de las llamas y permitan tomar acciones preventivas.

### ***3.5.1.2. Autonomía del dispositivo en el campo***

Capturar imágenes en zonas áridas e inhóspitas demanda una serie de requisitos físicos y técnicos para lograr el correcto funcionamiento, entre estos se destacan los siguientes factores:

#### **Resistencia del clima**

Los equipos deben contar al menos con protección IP65, de tal manera que pueda estar a la intemperie, manteniendo sus estructura y eficiencia intacta.

#### **Consumo energético**

Debido a la extensión de los terrenos, una red, tanto eléctrica como de telecomunicaciones cableada es una implementación inviable, ya que la distancia entre puntos de control (ya sean de CCTV o sensores) deberán superar 1 km entre sí, por lo cual se propone el uso de equipos autónomos de la red eléctrica que se comuniquen mediante radiofrecuencia, pudiendo ser WAP o WLAN.

#### **Batería, paneles solares e inversores**

El uso de fuentes de energía sustentables permite posicionar puntos de monitoreo autónomos a lo largo de un terreno, con el uso de combinado de panel solar, inversor y baterías, puede generar la energía suficiente y respaldo energético para mantener en servicio una cámara o sensor destinados a la captura de información para detectar posibles incendios.

Existen soluciones integradas en el mercado ya asociadas a la video vigilancia en áreas despobladas o con acceso limitado a la electricidad. Entre dichas soluciones se proponen dos proveedores nacionales y uno internacional:

- Cámara solar Hikvision: consiste en una cámara integrada con sistema de batería, panel solar y comunicación vía wifi y 4G. Esta solución está disponible en Argentina.



Figura 49: cámara solar – Hikvision

Fuente: (Hikvision. 2019)

- PowerCube 500, Huawei Site Power: consiste en un sistema integrado destinado a la videovigilancia, iluminación y/o comunicaciones. Esta solución consiste en un mástil de 4, 6, 8 o 10 metros, en cual se posiciona un arreglo de paneles solares, batería, y cámara. Esta solución también está disponible en Argentina.



Figura 50: Power Cube – Huawei

Fuente: (Huawei. 2018)

- CamSAT provee una solución integrada de paneles solares, batería y fuente DC para alimentar cámaras de CCTV. Esta solución está disponible en Europa.



Figura 51: iCAM-Solar365 – CamSAT

Fuente: (Camsat. 2020)

### ***3.5.1.3. Protocolos de comunicación***

Un aspecto importante para tener en cuenta a la hora de implementar una red de dispositivos IoT, independientemente si es de cámaras, sensores o una combinación de ambos, es el protocolo de comunicación que se va a elegir para que los dispositivos se puedan “hablar” entre sí. Ya que dependiendo de factores como: la distancia de la conexión, la disponibilidad o el tipo de dispositivos, va a convenir elegir un protocolo sobre los demás.

A continuación, se mencionarán algunos de los protocolos de IoT más conocidos dependiendo si es una para implementar en una red de corto o largo alcance.

### Redes de corto alcance y bajo consumo

Las redes de baja potencia y corto alcance están indicadas para hogares, oficinas y entornos de tamaño reducido. Normalmente, necesitan baterías capaces de suplir una carga inferior a los 50 vatios y su uso es popular en el mercado.

Algunos de los protocolos de IoT utilizados en estos casos son (Signals IoT. 2020):

- Zigbee: funciona creando una red de área local de malla. Este protocolo funciona bien para intercambios de datos poco frecuentes, a bajas velocidades y a distancias cortas (por ejemplo, en casas o edificios). Su especificación está basada en el protocolo IEEE 802.15.4. Este protocolo se caracteriza por tener un bajo costo de implementación, bajo consumo de energía y por su gran funcionamiento en envío de datos en tiempo real. Zigbee es un protocolo que ha ganado popularidad después de que en el 2019 se creara el proyecto “Connected Home over IP” que es una alianza de Apple, Google y Amazon y que busca desarrollar y promover la adopción de estándar de conectividad libre de regalías para aumentar la compatibilidad entre los dispositivos IoT para el hogar. Este proyecto va a adoptar el protocolo Zigbee, potenciando así este estándar en el mercado.
- Z-Wave: es un protocolo de comunicaciones utilizado principalmente para domótica. Funciona con una red en malla que usa ondas de radio de baja potencia para la comunicación de dispositivo a dispositivo. Las conexiones individuales tienen mayor alcance que las de Zigbee, pero tienen un mayor consumo energético y la señal solo puede hacer tres saltos.
- Wifi/802.11: es la opción más utilizada hoy en día para IoT en entornos hogareños y de oficina. Los dispositivos IoT Wifi consumen mucha más energía que los dispositivos con otros protocolos más eficientes, por lo que es posible que no sea la opción más adecuada para todos los escenarios.

### **Redes de área extensa de bajo consumo (LPWAN)**

Las redes de área amplia de baja potencia (conocidas como LPWAN, por sus siglas en inglés) permiten la comunicación en un radio desde 500 metros y hasta 10 kilómetros, tienen un consumo de energía mínimo (consumen 600 nA de corriente de reposo y 4.6 mA de corriente de recepción activa) y se usan para la mayoría de los dispositivos IoT.

Algunos de los protocolos de IoT LPWAN utilizados comercialmente son (Signals IoT. 2020):

- LoRa: es un protocolo de largo alcance para dispositivos inalámbricos IoT. Las señales pueden atravesar obstáculos y viajar a través de largas distancias gracias a los chips LoRa y a una estrategia de espectro ensanchado que puede transmitir a través de varias frecuencias y velocidades de datos.

Como referencia, este es el protocolo implementado en el proyecto “Zeroxygen - Track this fire” desarrollado por estudiantes de ingeniería en informática de la UBA en el 2020 (Schulkin. 2020)

- Sigfox: es un protocolo que utiliza las bandas de frecuencias no licenciadas ISM (se pueden utilizar para fines industriales sin previa autorización). Utiliza la tecnología UNB (siglas de banda ultra estrecha en inglés) que es capaz de transmitir datos a velocidades de 10 a 1.000 bits por segundo, utilizando muy poca potencia. Consume solo 50 micro vatios, 100 veces menos que lo que consumen las conexiones móviles.
- 4G LTE para IoT: estas redes ofrecen alta capacidad y baja latencia, por lo que son una opción ideal para escenarios de IoT que requieren envío de información en tiempo real.

#### ***3.5.1.4. Tipos de procesamiento***

Existen distintos enfoques a la hora de hablar de procesamiento de información capturada por dispositivos IoT: cloud computing y Edge computing (computación en la nube y computación en el borde, en castellano).

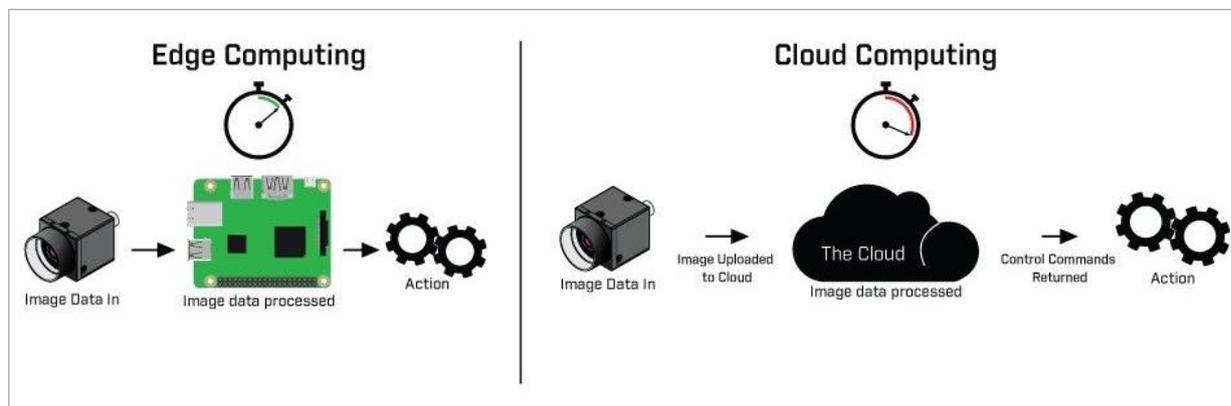


Figura 52: representación de edge computing vs cloud computing

Fuente: (Teledyne FLIR. 2022)

### Cloud computing

El Cloud computing es un enfoque en el cual una empresa/usuario hace uso de los recursos computacionales de un proveedor de servicio de nube en contraposición a tener que instalar servidores de manera local.

La computación en la nube ofrece a los usuarios y empresas la posibilidad de acceder a distintos servicios provistos por el proveedor bajo demanda, disminuyendo así costos de mantenimiento, configuración, etc. El usuario solo requiere de una conexión a internet para poder acceder a dichos servicios.

Algunos de los beneficios que se obtienen con este enfoque son (Yeung, 2022):

- Menor costo inicial: permite a las empresas lanzar aplicaciones al mercado rápidamente, con una barrera financiera de entrada baja, ya que no requieren de compra de hardware, administración de TI, electricidad las 24 horas para energía y refrigeración, etc.
- Precios flexibles: solo se le paga a proveedor de la nube por los recursos utilizados, lo que permite un mayor control sobre los gastos.
- Cómputo bajo demanda ilimitado: el cloud computing les da a los usuarios la libertad de adaptarse a las demandas cambiantes al instante mediante el aprovisionamiento y desaprovisionamiento automático de recursos, lo que permite aumentar la eficiencia general de las organizaciones.

- Gestión de IT simplificada: los proveedores de la nube brindan acceso a expertos en gestión de IT, lo que permite a la empresa concentrarse en las necesidades principales de su negocio.
- Actualizaciones sencillas: se puede acceder al último hardware, software y de manera casi instantánea.
- Confiabilidad: los datos se pueden duplicar en múltiples sitios redundantes en la red del proveedor de la nube.
- Ahorro de tiempo: las empresas pueden implementar aplicaciones en una fracción del tiempo y llegar al mercado antes, sin invertir tiempo en tareas de configuración de redes y servidores privados.

### **Edge computing**

A diferencia del Cloud computing, el enfoque de Edge computing pretende darle autonomía al dispositivo a través de permitir el procesamiento de los datos desde el dispositivo que los capturó (de aquí el término “en el borde”) en comparación con hacerlo desde el servidor central Cloud.

El Edge computing permite un procesamiento de datos más rápido, así como un mayor ancho de banda en la red, porque los datos no necesitan viajar hasta un servidor central, sino que con que llegue el resultado del procesamiento, es suficiente para tomar acciones.

Algunos de los beneficios que se obtienen con este enfoque son (Yeung, 2022):

- Latencia más baja: debido a que se reducen la cantidad de datos que viajan por la red.
- Costo reducido: el uso de la red de área local para el procesamiento de datos otorga a las organizaciones un mayor ancho de banda y almacenamiento a costos más bajos. Adicionalmente impacta en un menor costo en la cantidad de datos que viajan por la red.
- Precisión del modelo de inteligencia artificial: al tener una red donde no viajan datos de más (por ejemplo, imágenes donde claramente no hay un incendio), se alivia la red. Al punto tal que se evitan problemas relacionados con una red colapsada. Esto ayuda a que se puedan ejecutar varios modelos de IA de manera simultánea, mejorando los bucles de retroalimentación de datos que se pueden usar para mejorar la precisión del modelo.

- 
- Alcance más amplio: como se pueden procesar datos de manera local, se puede ampliar el alcance de la red a ubicaciones donde no hay conexión a internet.
  - Soberanía de datos: el Edge computing permite mantener los datos confidenciales y dentro de la red de área local. Lo que disminuye la exposición a ataques de ciberseguridad y un mejor cumplimiento de las leyes de datos estrictas (como GDPR, en Europa) y en constante cambio.

### 3.5.1.5. Costos de implementación

En cuanto a los costos para tener en cuenta a la hora de implementar una solución de machine learning basada en el análisis de imágenes para detectar incendios, se plantean tres escenarios de referencia:

#### Cámaras para exteriores

El primer escenario propone utilizar cámaras de vigilancia para exteriores que permita transmitir vía streaming una foto de la zona cada 30 minutos.

Para este caso, se pueden utilizar las cámaras de videovigilancia para exteriores DS-2CD2T87G2-LSU/SL marca Hikvision (disponibles en la Argentina). Esta cámara cuenta con una resolución de 8 megapíxeles, alimentación PoE (Power over Ethernet, en inglés), protección IP67 (resistente al agua y al polvo). Además, soporta temperaturas entre -30 y 60 °C y humedad relativa en el ambiente del 95%, lo que significa que puede estar 24 horas a la intemperie sin riesgo a sobrecalentamiento. Cuenta con una apertura horizontal de 102° y vertical de 52°, brindando un amplio campo de vision.



Figura 53: cámara Hikvision DS-2CD2T87G2-LSU/SL

Fuente: (Hikvision. 2020)

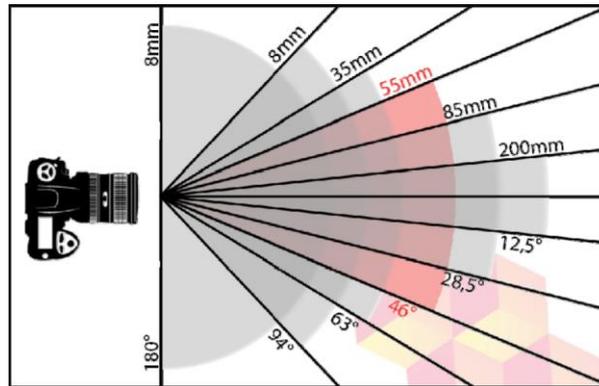


Figura 54: representación del ángulo de visión según la distancia focal

Fuente: (Barba. 2014)

La cámara de exteriores Hikvision tiene un costo de 770 USD. Al ser una cámara estanca, no requiere de inversión adicional para poder utilizarla en un ambiente inhóspito.

Hikvision ofrece una garantía de 5 años para este tipo de cámaras, lo que avala su durabilidad y calidad en este caso de uso.

### Edge computing usando AWS DeepLens

Edge computing (computación en el borde, en castellano) es una filosofía que busca darle autonomía a los dispositivos IoT a través de analizar resultados de manera local en vez de usar los dispositivos de una red IoT solamente para capturar datos y enviarlos a la nube.

AWS cuenta con una cámara pensada para Edge computing llamada ‘AWS DeepLens’. Es una cámara de video para desarrolladores pensada para aplicar deep learning en vivo mientras se captura el video. DeepLens permite ejecutar modelos propios entrenados con Amazon SageMaker o utilizar modelos de una biblioteca de modelos pre-entrenados de AWS.



Figura 55: AWS DeepLens (edición 2019)

Fuente: (AWS. 2019)

AWS DeepLens (edición 2019) tiene un costo de 249 USD. Al no ser una cámara de exteriores, requerirá de una caja estanca para protegerla en la intemperie, agregándole 20 USD más al costo estimado por cámara.

### **AWS Panorama**

AWS Panorama es un servicio de Amazon que permite aplicar computer vision a una red de cámaras IP local.

Para poder analizar imágenes en vivo usando modelos de machine learning en AWS Panorama, se necesita de un dispositivo llamado ‘AWS Panorama Appliance’ (u otro dispositivo similar de otras marcas) al cual se le van a conectar las cámaras IP y es el encargado de procesar las imágenes capturadas por la cámara y analizarlas con el modelo previamente entrenado. A su vez, AWS Panorama Appliance se puede integrar con otras aplicaciones on-

premise o en la nube para automatizar procesos y tomar acciones en función de los resultados obtenidos del análisis de las imágenes.

En la figura 56 se puede observar un diagrama de cómo se integra AWS Panorama con AWS Panorama Appliance, la red de cámaras IP y aplicaciones on-premise/cloud.

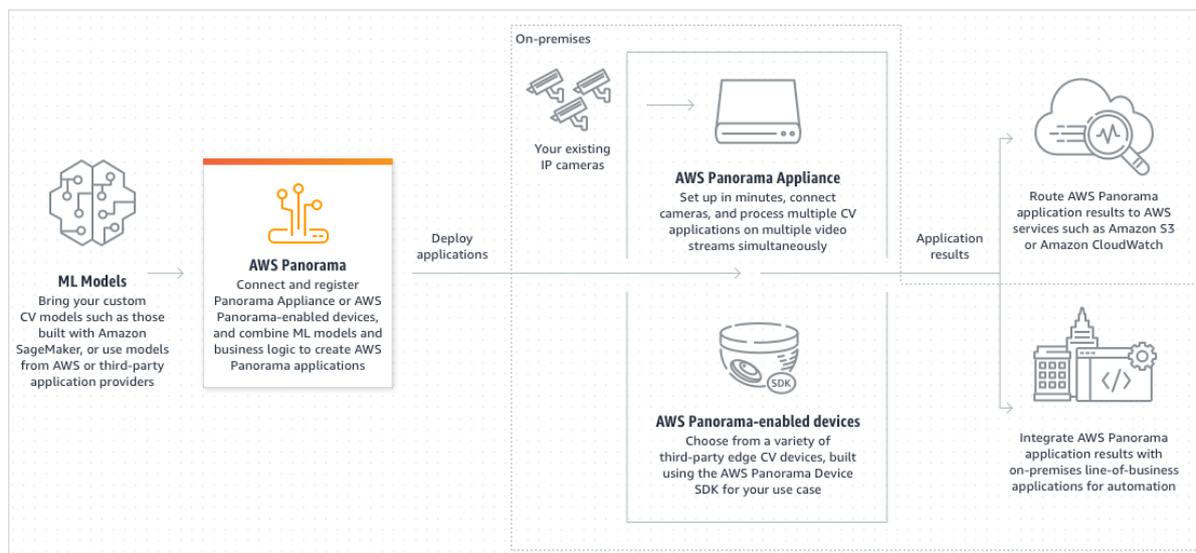


Figura 56: flujo de AWS Panorama

Fuente: (Amazon Web Services. 2022)

AWS Panorama Appliance tiene un costo de 3.999 USD, por lo que si bien, se puede reutilizar una red de cámaras comunes ya instaladas en la zona que se desee monitorear, requiere de una inversión de ~4 mil dólares estadounidenses para poder aprovechar los servicios de computer vision de AWS.

### 3.5.2. Integración con los principales Cloud Providers en el mercado

Cuando se elige un proveedor de la nube para desplegar una red de dispositivos IoT se deben tomar en cuenta los servicios y soluciones del internet de las cosas que dicho proveedor pueda ofrecer.

Los servicios de IoT se pueden clasificar en los siguientes grupos (Hasan. 2022):

- Gestión/habilitación de aplicaciones IoT: esta categoría está conformada por los servicios diseñados para permitir que desarrolladores creen y administren aplicaciones de IoT.
- Gestión de dispositivos IoT: incluyen servicios en la nube diseñados para garantizar que los dispositivos conectados funcionen correctamente. Algunos ejemplos incluyen monitoreo de dispositivos, actualizaciones de firmware o administración de configuración de implementación.
- Gestión/habilitación de datos de IoT: estos servicios están diseñados para una conectividad fluida de dispositivo-borde y para proporcionar capacidades para almacenar y analizar datos de IoT.

Este PFI se va a centrar en los 3 principales Cloud Providers: Amazon Web Services, Microsoft Azure, y Google Cloud.

### ***3.5.2.1. Amazon Web Services***

AWS fue el primer proveedor de nube en ofrecer soluciones de IoT al público en general.

La ventaja sobre los otros proveedores es que ofrece una mayor cantidad de servicios específicos de IoT. Según Hasan (2022), el servicio de gestión de datos IoT llamado AWS IoT Core es el servicio más popular en el mercado (basado en estudios de casos públicos de IoT).

AWS cuenta con 13 servicios en la nube dirigidos a IoT:

- Gestión/habilitación de aplicaciones IoT: AWS IoT TwinMaker, AWS IoT Events, AWS IoT Roborunner, AWS IoT FleetWise
- Gestión de dispositivos IoT: AWS IoT Device Manager, AWS IoT 1-Click
- Gestión/habilitación de datos de IoT: AWS IoT Core, AWS IoT Analytics, AWS IoT SiteWise, AWS IoT Greengrass
- Otros servicios en la nube de IoT: AWS IoT Device Defender, Free RTOS, y AWS IoT ExpressLink

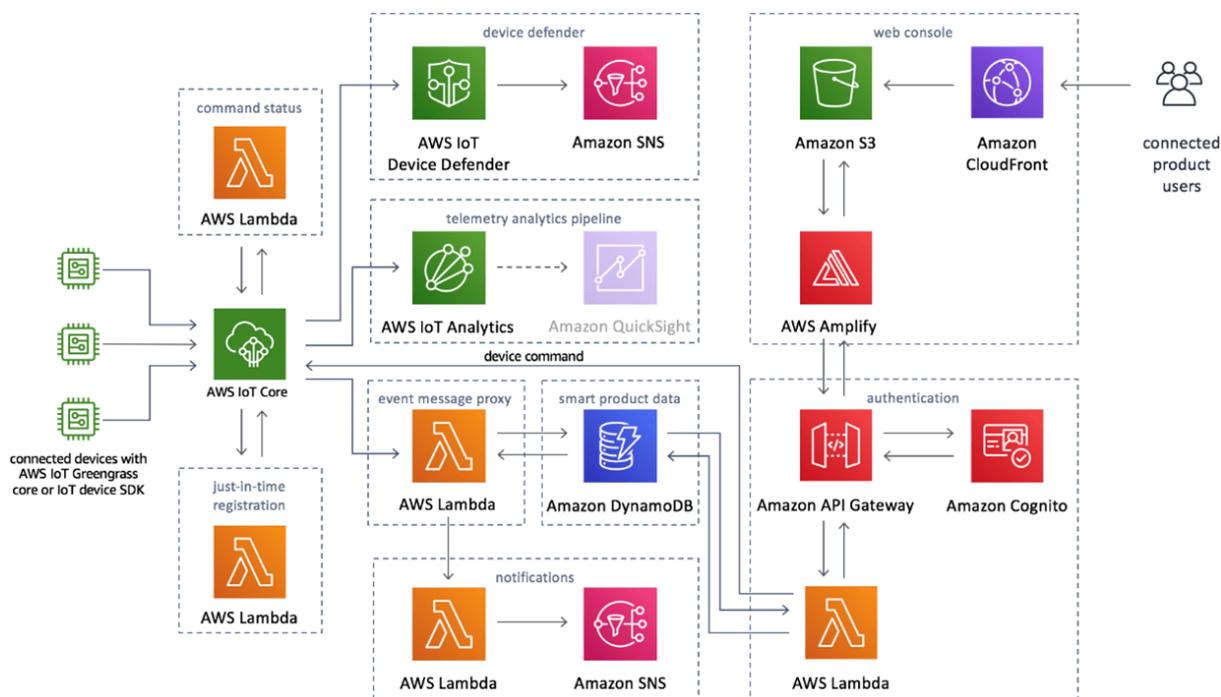


Figura 57: ejemplo de arquitectura AWS para productos inteligentes

Fuente: (AWS. 2020)

### 3.5.2.2. Microsoft Azure

Microsoft tiene un enfoque específico en IoT a nivel industrial y desde el Edge computing. El diferenciador de Microsoft Azure es su integración con herramientas de business intelligence (como Power BI), sus soluciones específicas de la industria (por ejemplo, Microsoft Cloud for Retail o Microsoft Cloud for Manufacturing) y el fuerte enfoque de la empresa en el soporte empresarial (Hasan. 2022).

Microsoft Azure cuenta con nueve servicios en la nube dirigidos a IoT:

- Gestión/habilitación de aplicaciones IoT: Azure Digital Twins y Azure IoT Central
- Gestión de dispositivos IoT: Azure IoT Hub
- Gestión/habilitación de datos de IoT: Azure IoT Edge, Azure Percept y Azure Time Series Insights
- Otros servicios en la nube de IoT: Azure Sphere, Azure RTOS, y Windows 10 IoT Core Services

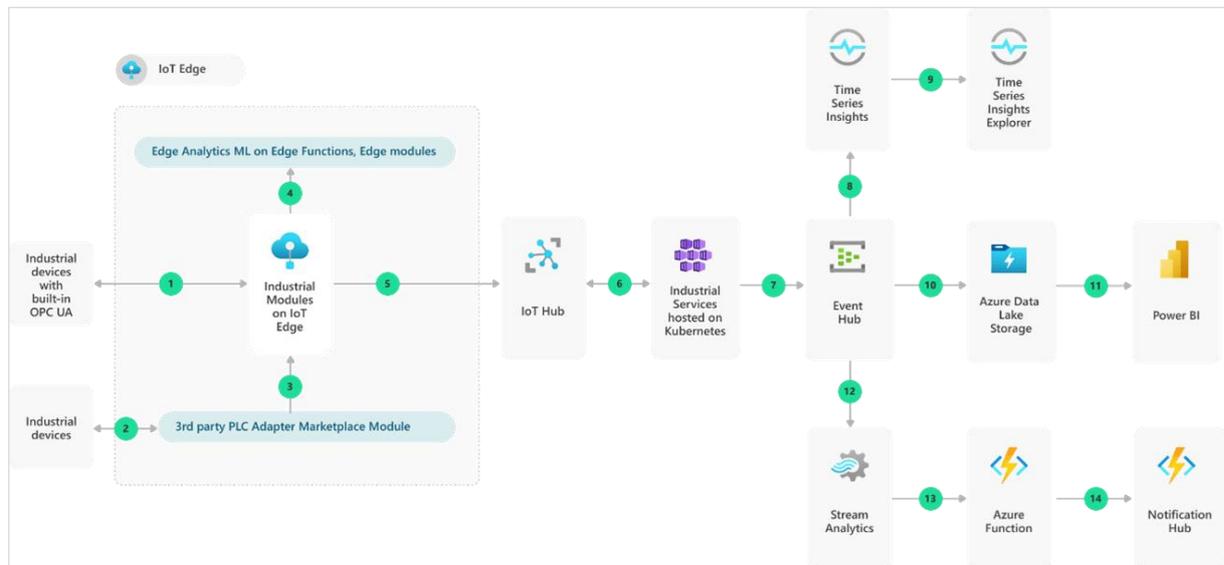


Figura 58: ejemplo de arquitectura unificada de comunicación de dispositivos IoT con servicios de Azure

Fuente: (EdPrice-MSFT. 2022)

### 3.5.2.3. Google Cloud

Google cuenta con más de 100 servicios en la nube, pero solo uno de ellos está dedicado 100% a IoT: IoT Core. Este servicio es muy similar al IoT Hub de Microsoft que se utiliza para conectar dispositivos IoT a la nube de Google. Se puede combinar con otros servicios ofrecidos por Google Cloud para crear una solución de IoT de extremo a extremo.

A diferencia de sus competidores Azure y AWS, este servicio no proporciona servicios de datos específicos de IoT (como, por ejemplo, el AWS IoT Analytics de Amazon).

Google Cloud IoT Core se retirará del mercado el 16 de agosto de 2023 (Gross. 2022), obligando a sus clientes a migrar a plataformas especializadas en servicios IoT.

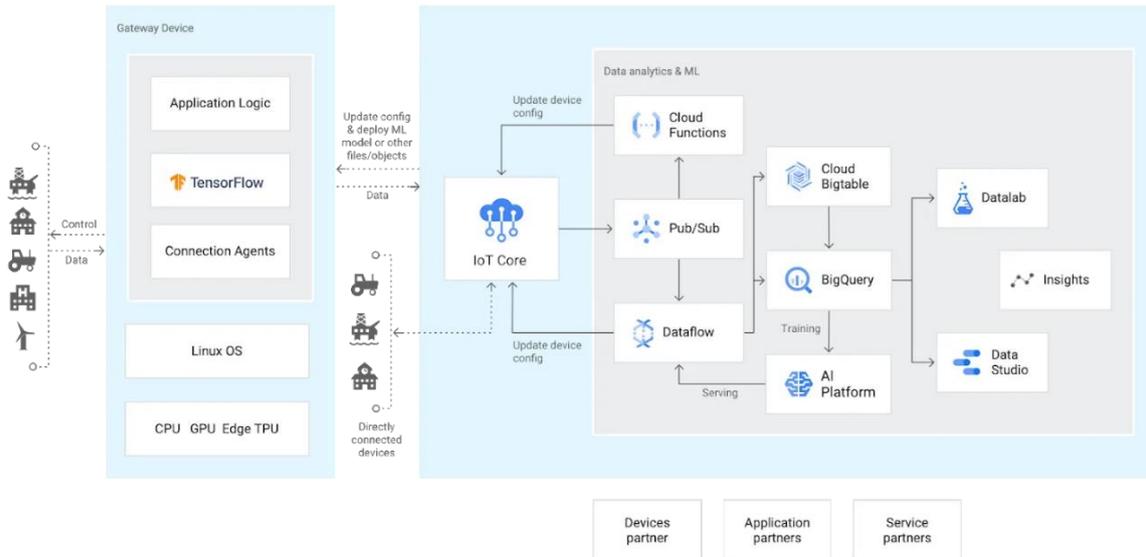


Figura 59: ejemplo de arquitectura en Google Cloud

Fuente: (Google Cloud. 2022)

### 3.5.3. Conclusiones

Cuando se habla de utilizar dispositivos IoT para combatir incendios forestales hay que apuntar a detectarlos de manera temprana y actuar lo antes posible para minimizar los daños generados.

Una red ideal es aquella que logra combinar información capturada por todos los distintos dispositivos que se definieron anteriormente (satélites, cámaras y sensores). Adicionalmente, al involucrar al humano a través de la aplicación móvil StopFire, que es el causante del 95% de los incendios que ocurren en Argentina, se logra atacar la causa raíz y crear consciencia en la sociedad con respecto a este tema.

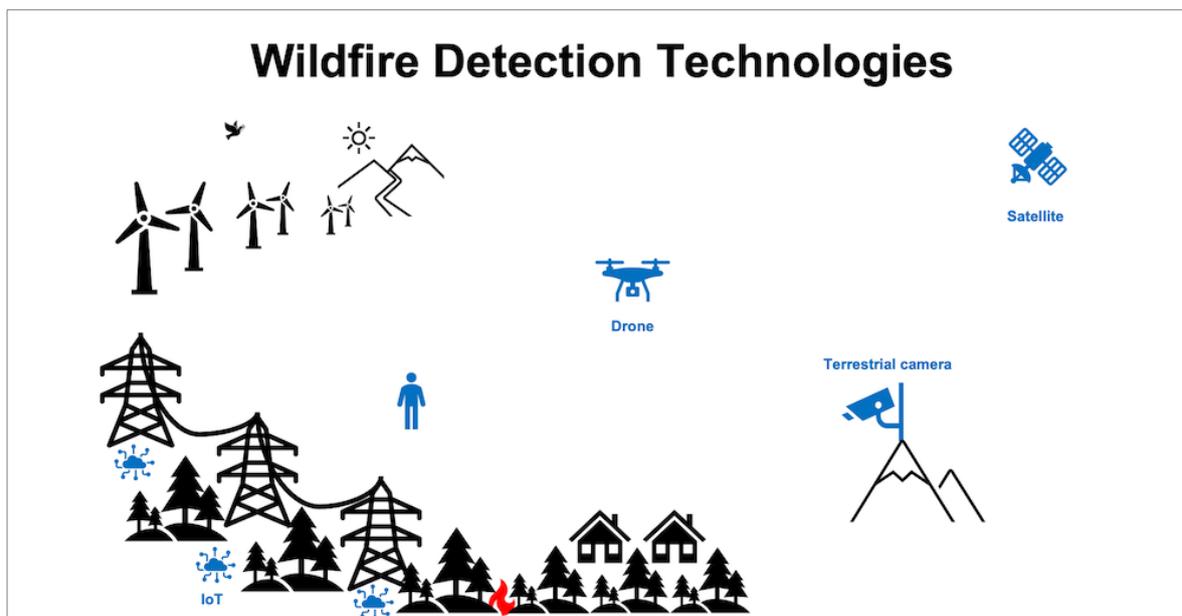


Figura 60: diagrama con tecnologías comunes de detección de incendios forestales

Fuente: (Bedolla. 2022)

Al ser las cámaras el dispositivo IoT central de este PFI, se realizó un análisis de alcance visual entre dos puntos simulado mediante la herramienta ISP Design Center de Ubiquiti (ver figura 61), que permite identificar puntos entre los cuales no existe interferencia topográfica, de esta manera se puede garantizar el alcance visual y de comunicación mediante radio enlaces que satisfacen la necesidad visual y de comunicación entre los dispositivos IoT y una base central o punto de control desde la cual se podrá enviar la información hacia la nube para poder ser analizada y accionada.

Una ruta eficiente se considera aquella en la cual se requieren de la menor cantidad de puntos de interconexión para lograr cubrir la mayor cantidad de terreno vigilado mediante el sistema StopFire.

Elegir una ruta eficiente permitirá elaborar el esquema más económico, pero requiere de un relevamiento topográfico previo a la implementación y al cálculo del sistema para estimar el rango de cobertura, teniendo en cuenta los principales focos de inicios de incendios (por ejemplo, si la mayoría de los incendios son causa de la naturaleza humana, incluir dentro de la ruta eficiente las rutas nacionales, centros de turismo, etc.)

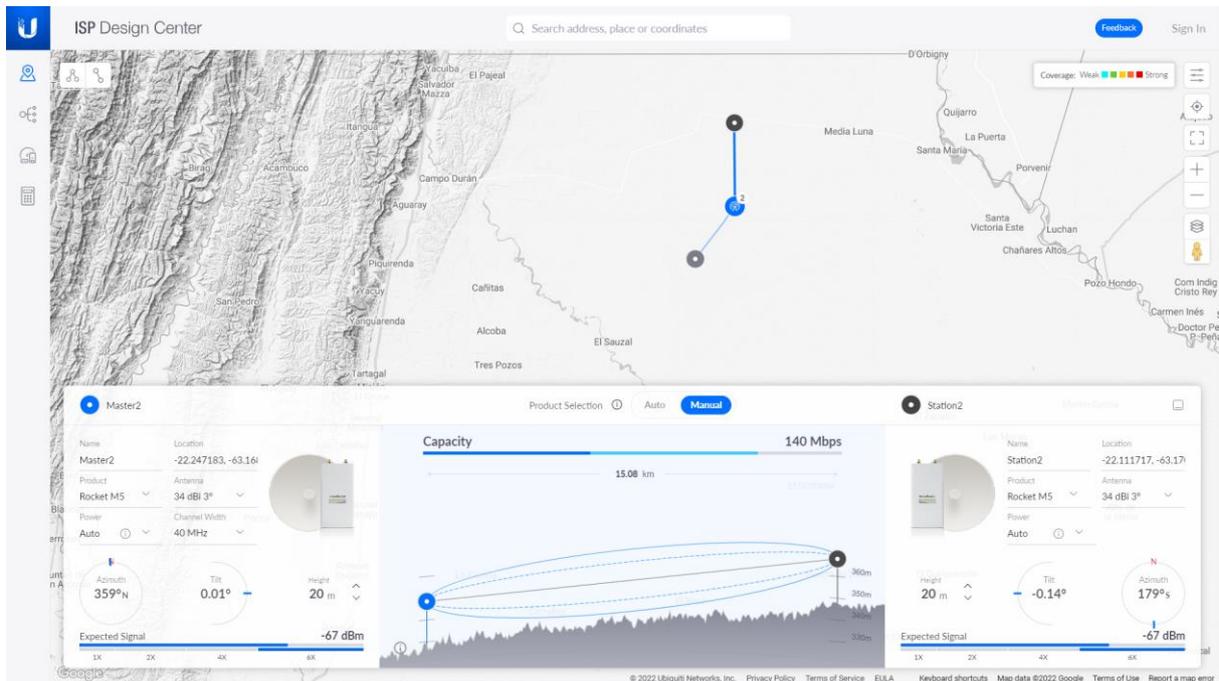


Figura 61: análisis de radio enlace (Formosa, Argentina) usando Ubiquiti ISP Design Center

Fuente: elaboración propia

¿Dónde StopFire aporta su valor? Indiferentemente de la fuente de la imagen, la CNN desarrollada para ese PFI va a poder analizarla y generar una alerta en caso de que exista fuego. El modelo permite analizar imágenes aéreas (drones), desde un ángulo recto (cámaras IP fijas instaladas en la zona), o desde un ángulo contrapicado (tomadas con un teléfono celular por un peatón en la zona del incendio).

StopFire se basa en imágenes capturadas con cámaras, por lo que el uso de imágenes satelitales queda fuera del alcance de este PFI.

#### 4. Metodología y desarrollo

Este PFI se desarrolló llevando a cabo metodologías ágiles. Lo que se busca con la implementación de esta metodología es poder mostrar resultados de manera rápida y efectiva ya que uno de los principios ágiles es el tener software funcionando después de cada sprint. Permitiendo así el avance iterativo e incremental del producto desarrollado.

Se eligió un marco de trabajo ágil, iterativo e incremental porque facilita la colaboración entre participantes y porque da flexibilidad a la hora de adaptar cambios sobre producto ya entregado que aún tiene espacio para mejorar.

Se definieron sprints de 1 semana y, al culminar la semana, se tenía una revisión de las tareas realizadas, cuáles faltaban completar, y cuales se iban a priorizar para el siguiente sprint. De esta forma, se logró generar un avance real del proyecto semana a semana.

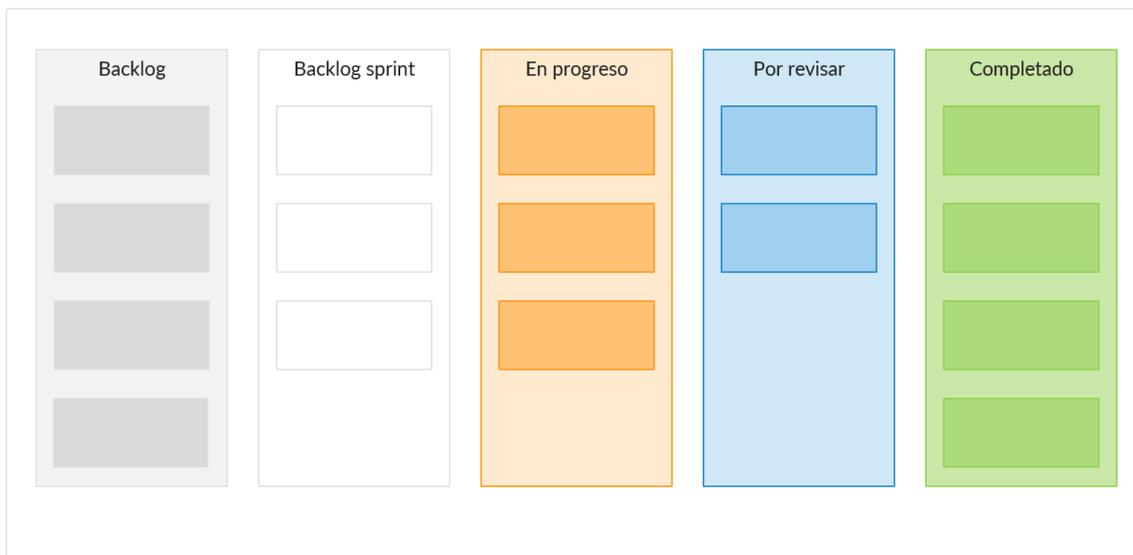


Figura 62: modelo del framework ágil utilizado en este PFI

Fuente: elaboración propia

#### 4.1. Herramientas utilizadas

Las herramientas de desarrollo y documentación utilizadas para este PFI fueron:

- Google Colab: es un servicio cloud de Jupyter notebooks creado por Google que permite escribir y ejecutar código en Python. Eliminando la necesidad de descargar, instalar y ejecutar localmente las librerías que se deseen implementar en el modelo.
- Visual Studio Code: es una IDE creada por Microsoft que permite escribir código fuente en texto plano y ofrece herramientas avanzadas para programadores como, por ejemplo, integración con Git.
- GitHub: es una plataforma de desarrollo colaborativo de software que permite almacenar proyectos utilizando el sistema de control de versiones Git diseñado por Linus Torvalds.
- Flutter: es un SDK desarrollado por Google que permite desarrollar aplicaciones nativas para móviles, web y de escritorio usando una única base de código.
- Amazon S3: es el servicio de almacenamiento de objetos de AWS que ofrece centralizar archivos y datos en la nube de manera segura y escalable.
- Amazon EC2: es un servicio web de AWS que ofrece computación en la nube de tamaño modificable, ajustándose a la carga de trabajo de manera dinámica.
- MySQL: es un sistema de gestión de bases de datos relacionales de código abierto.
- Heroku: es una plataforma de servicios en la nube que permite desplegar aplicaciones web desarrolladas en Node.JS, Ruby, Java, PHP, Python, entre otros lenguajes.
- Postman: es una herramienta que permite crear peticiones sobre APIs. Lo que facilita el testeado de endpoints de aplicaciones backend basadas en API REST.
- Moqups: es una plataforma web colaborativa que permite crear y validar prototipos funcionales de sitios web o aplicaciones móviles.
- Hugging face: plataforma web que permite crear, entrenar, e implementar modelos de machine learning.
- Trello: es una herramienta para gestión de proyectos y visualización de tareas que utiliza un tablero, listas y tarjetas para manejar el flujo del trabajo.

#### 4.2. Frameworks, lenguajes, algoritmos y librerías

Los frameworks, lenguajes de programación, algoritmos y librerías utilizados en el desarrollo de StopFire fueron:

- Python: es un lenguaje de programación de propósito general que permite el desarrollo de sitios webs, de software, análisis y visualización de datos, entre otros.
- Fast.ai: es una librería de Deep learning basada en PyTorch y desarrollada por Jeremy Howard y Rachel Thomas que facilita la creación y entrenamiento de redes neuronales.
- Flask: es un micro framework de Python para el desarrollo de aplicaciones web que permite el desarrollo de aplicaciones robustas y escalables.
- Streamlit: es un framework de código abierto en Python que facilita la creación de aplicaciones web para machine learning y data science.

Se utilizó para el desarrollo del front end de la aplicación web de StopFire.

Se eligió trabajar con ese stack debido al conocimiento que ya tenía el equipo gracias a desarrollos anteriores. Con respecto a Streamlit, los autores de este PFI no contaban con experiencia desarrollando con ese framework, pero después de investigar y confirmar con colegas expertos en desarrollo web se descubrió que la curva de aprendizaje era más rápida en comparación a otros frameworks, por lo que se eligió como el framework para el desarrollo del front end.

## 5. Pruebas realizadas

### 5.1. Red neuronal convolucional

Para probar el rendimiento de la red neuronal convolucional, se subió el archivo PKL con la versión final del modelo a la plataforma Hugging Face y se validó la performance del modelo analizando distintos tipos de imágenes.

Los escenarios probados se explican a continuación:

Neblina o niebla: este fenómeno aparece comúnmente en zonas húmedas del país. Si bien, es poco probable que se forme neblina en zonas desérticas o con baja humedad, igualmente podría ocurrir.

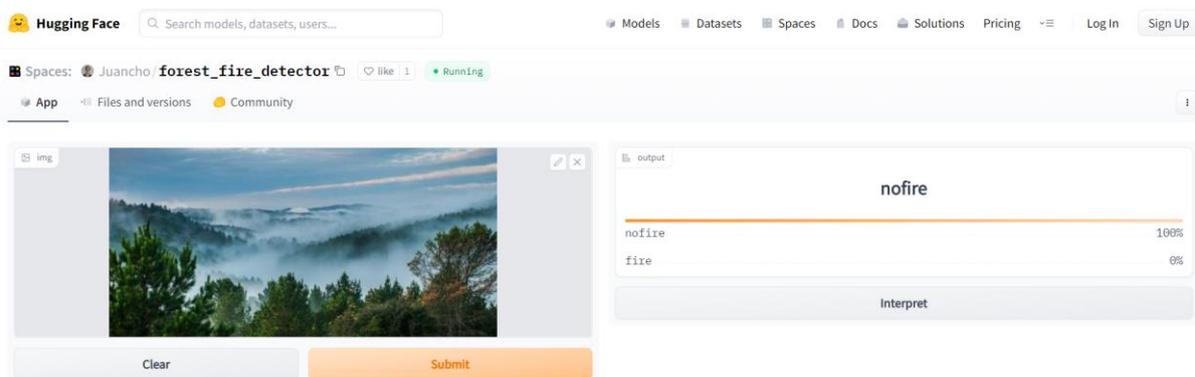


Figura 63: imagen #1 con neblina y resultados

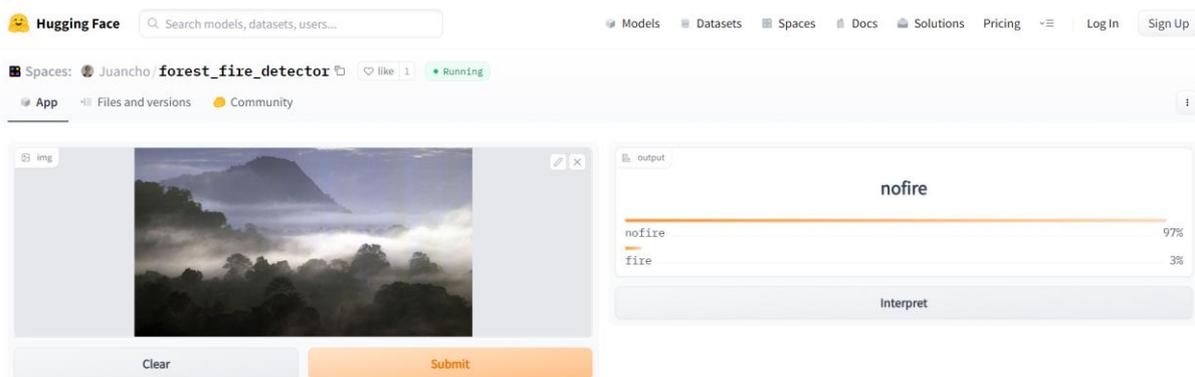


Figura 64: imagen #2 con neblina y resultados

Tormenta de arena: son comunes en la Pampa seca y en otras regiones áridas y semiáridas del territorio argentino.

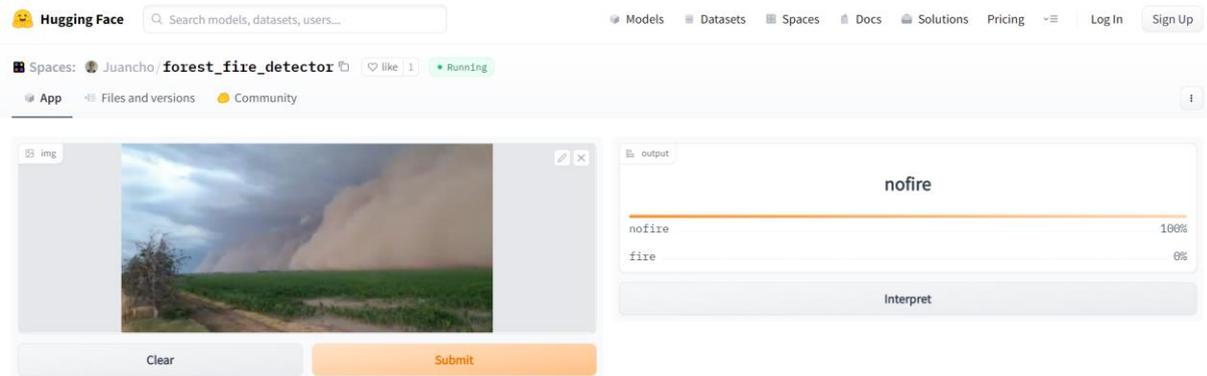


Figura 65: imagen #1 de tormenta de arena con resultados

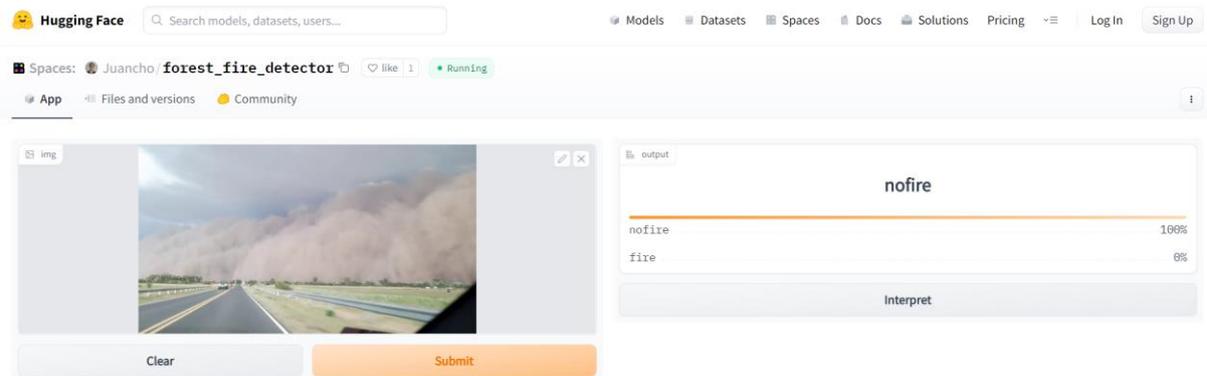


Figura 66: imagen #2 de tormenta de arena con resultados

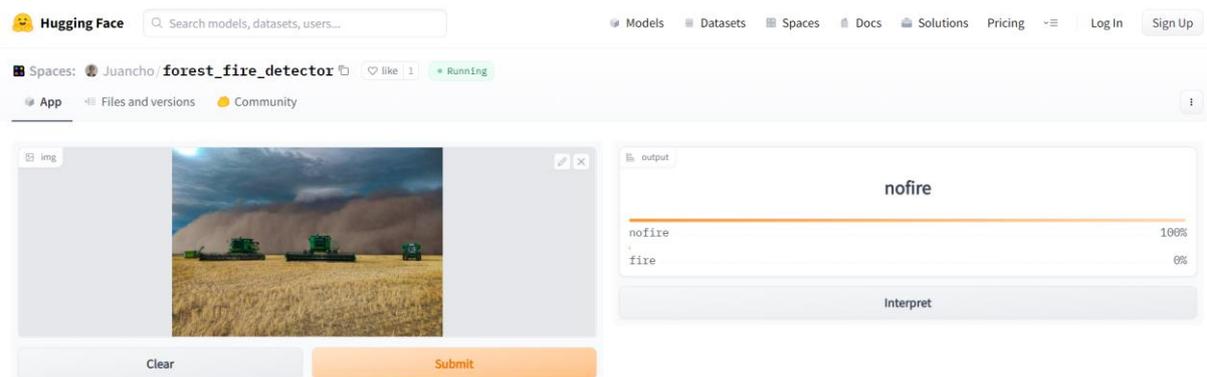


Figura 67: imagen #3 de tormenta de arena con resultados

Pastizales: se analizaron imágenes de pastizales puesto que en el sector de agricultura es una práctica tradicional prender en llamas pastizales (de manera intencional) para limpiar los campos para futuras siembras. Es un escenario propenso a tornarse en un incendio incontrolable que requiera la presencia de bomberos en la zona.

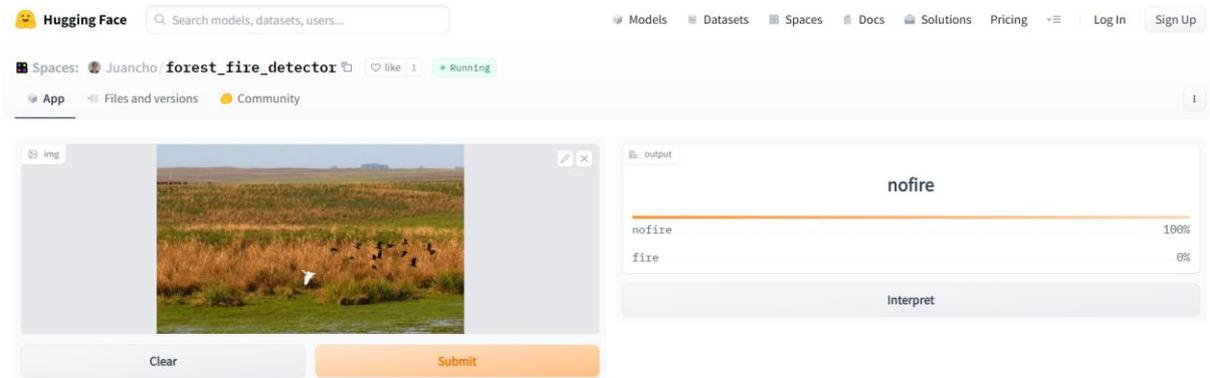


Figura 68: imagen de pastizal sin fuego con resultados

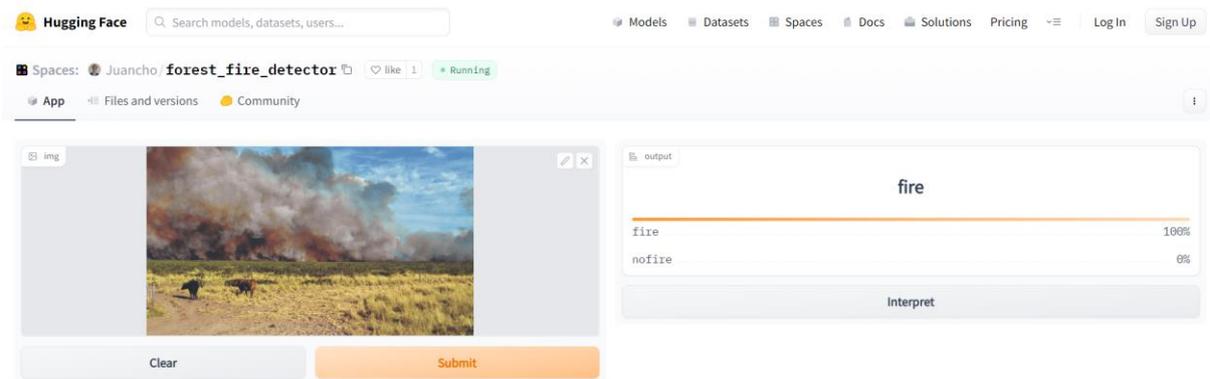


Figura 69: imagen de pastizal en llamas con resultado

También, se realizaron pruebas para los casos en los que la imagen capturó solamente presencia de columnas de humo, en la que se ve fuego y, por último, imágenes desde una vista peatonal simulando las fotos capturadas por los patrulleros.

Columnas de humo sin presencia de fuego

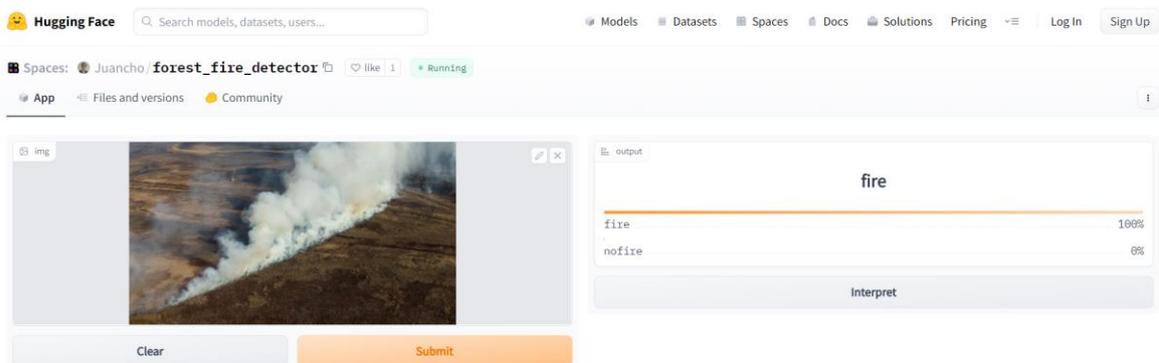


Figura 70: imagen de humo blanco con resultados

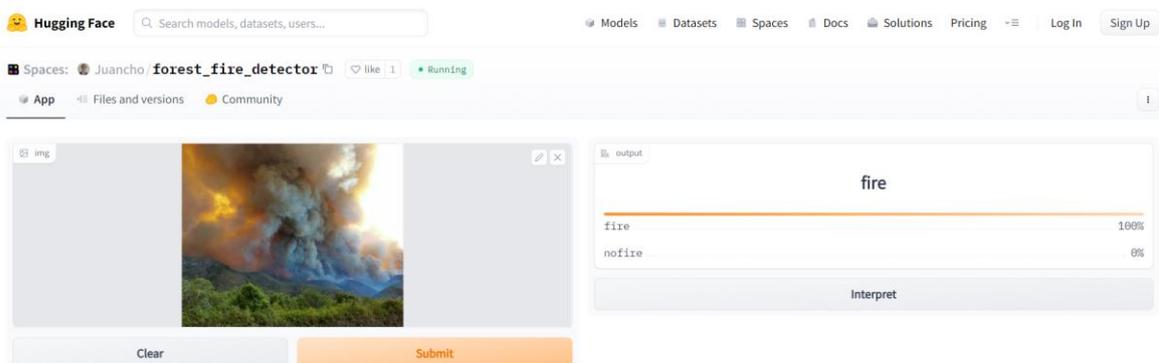


Figura 71: imagen de humo negro con resultados

Columnas de humo con presencia de fuego

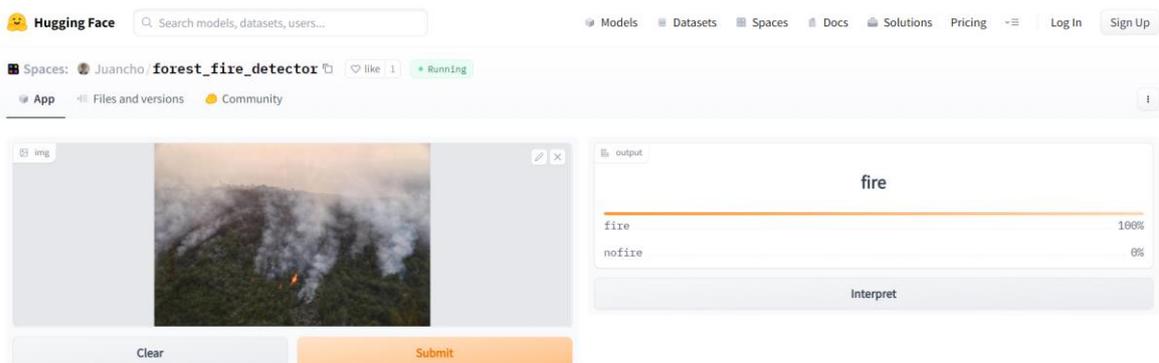


Figura 72: imagen con leve presencia de llamas

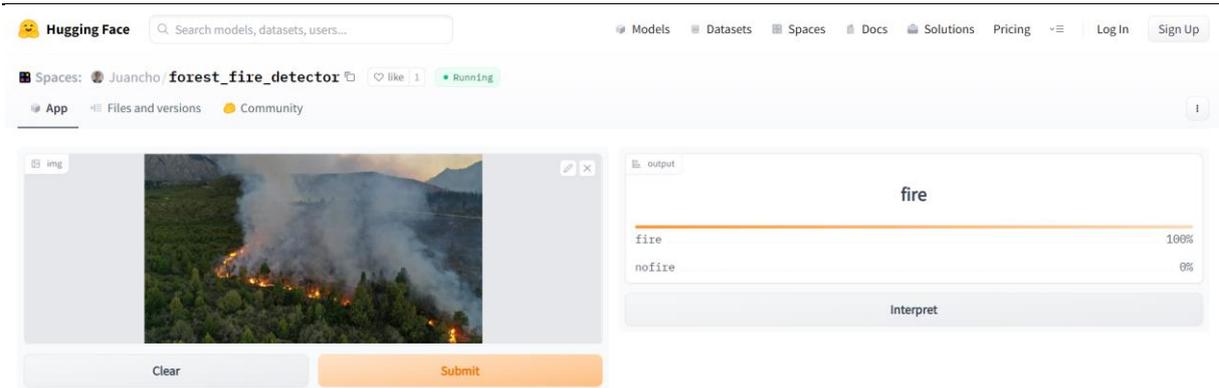


Figura 73: imagen con presencia media de llamas

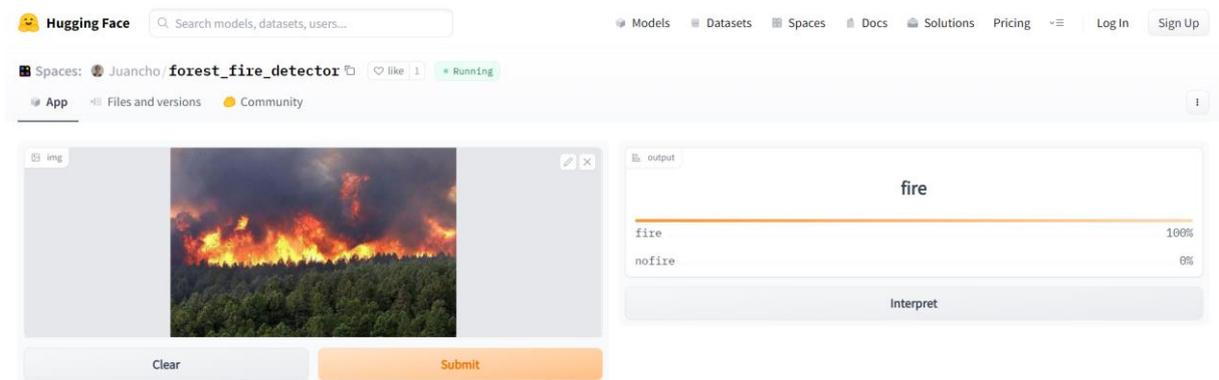


Figura 74: imagen con alta presencia de llamas

Fuego desde vista peatonal

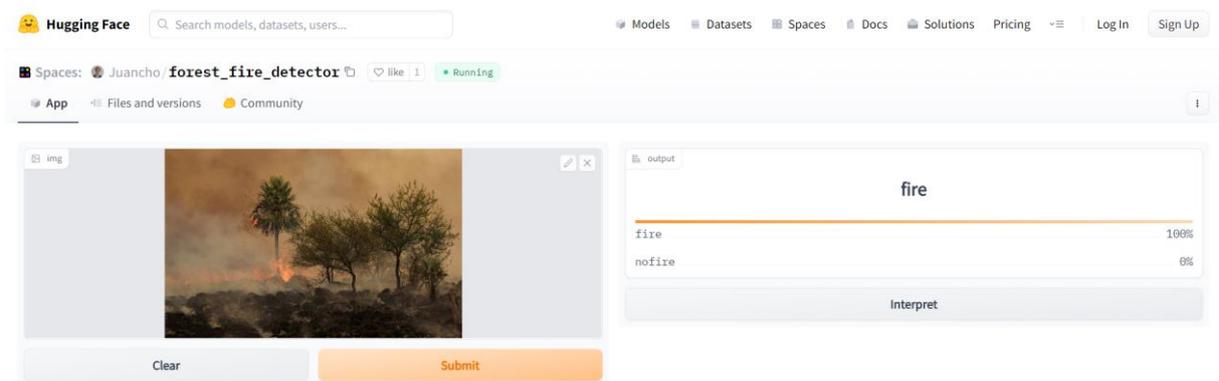


Figura 75: imagen #1 de incendio y humo desde una vista contrapicada con resultados

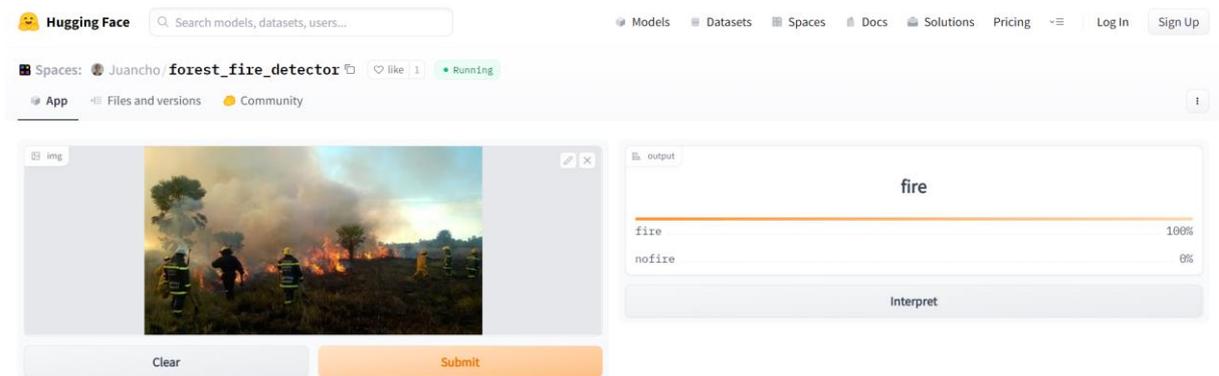


Figura 76: imagen #2 de incendio y humo desde una vista contrapicada con resultados

## 5.2. Envío de imagen y geolocalización desde la aplicación móvil

Se probó la aplicación móvil enviando una foto capturada en vivo y se verificó que la imagen fuese analizada por el modelo en el backend.

En caso de que el modelo detectara la existencia de fuego o humo, se validó que se enviara la imagen junto a la ubicación geográfica (latitud, longitud) correctamente y que apareciera en el dashboard de alertas de incendios.

Alerta enviada desde la aplicación móvil con fuego:



Figura 77: landing page de la aplicación móvil

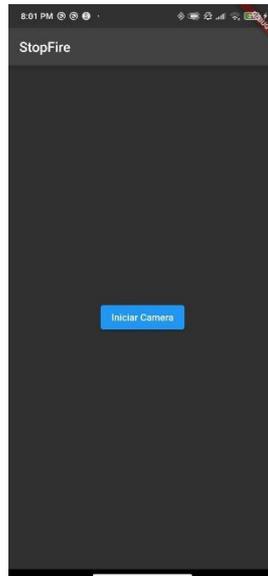


Figura 78: pantalla de inicio con el botón de “iniciar cámara” para realizar la alerta



Figura 79: cámara activada capturando una imagen desde la computadora para simular el incendio visto desde el celular



Figura 80: foto capturada desde la aplicación móvil



Figura 81: pantalla de confirmación una vez se envía la foto seleccionando el botón de “Enviar” en la pantalla anterior

Después de realizar la alerta, se validó que el backend recibiera correctamente la alerta (fig. 82)

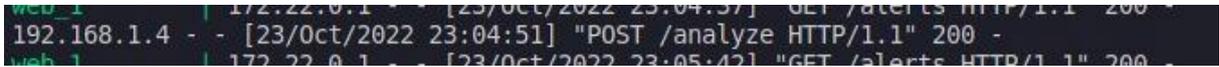


Figura 82: POST recibido en el backend

También se verificó que la alerta se guardara correctamente en la base de datos (fig. 83)

id	key	image	longitude	latitude	prediction	taken at	camera type	camera id	false alarm	zone	
1	14	18b751d4-53	http://forest-fire-dis	-58.398	-34.595	fire	2022-10-23 20:04:49	Punto móvil	App StopFire	0	Patrullero

Figura 83: base de datos actualizada con la alerta

Y por ultimo, se validó que la alerta apareciera correctamente en la aplicación web: imagen capturada, geolocalizacion y etiqueta de “punto movil” correctamente cargada.

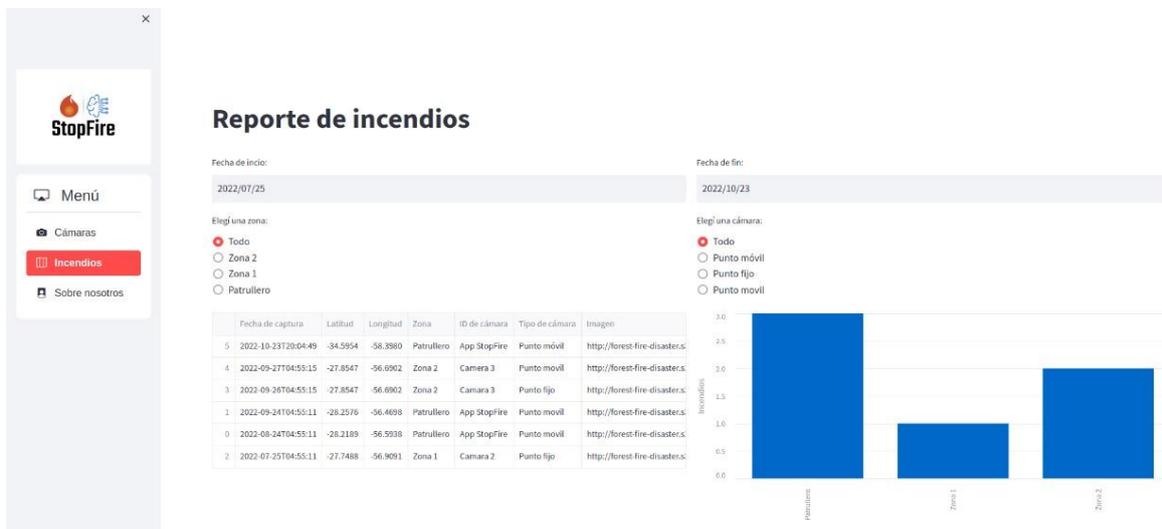


Figura 84: dashboard con las alertas desde el 25 de julio de 2022 al 23 de octubre del 2022

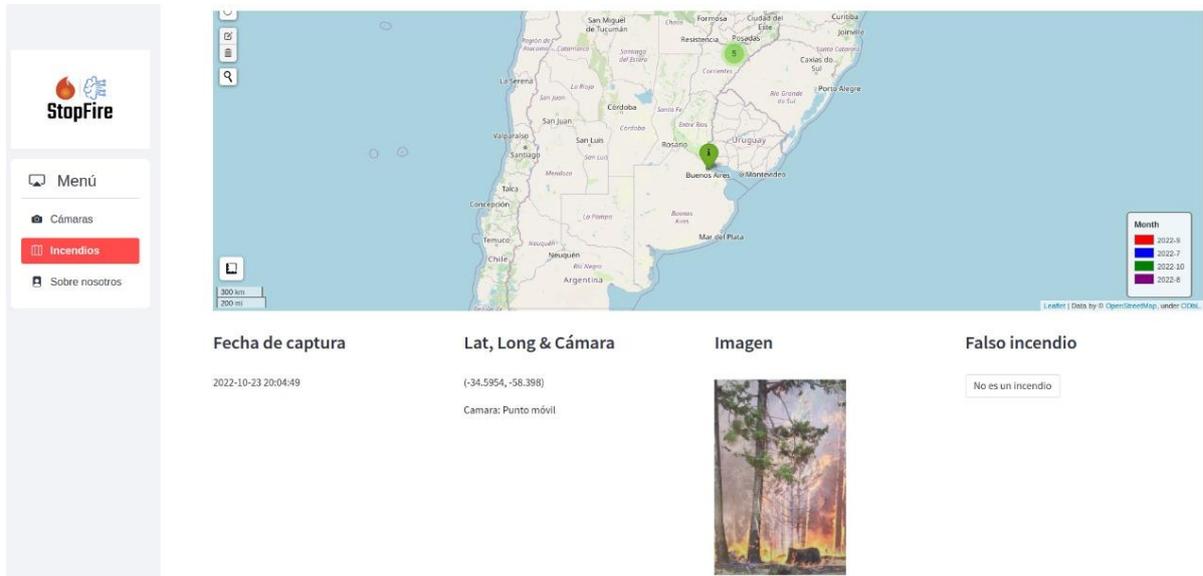


Figura 85: mapa de incendios con las alertas desde el 25 de julio de 2022 al 23 de octubre del 2022

También se probó el escenario en que se intenta enviar una la imagen y se reciba cualquier status que no sea el 200. Esto sucede en caso de que haya un error con el servidor o con la aplicación en sí y no se envía la imagen correctamente (fig. 86). En ese momento, la aplicación debe mostrar un mensaje de error indicándole al usuario que vuelva a realizar la alerta.



Figura 86: mensaje de error por fallo en el envío de la alerta

## 6. Branding

En esta sección se van a explicar dos aspectos importantes que distinguen y caracterizan la marca ‘StopFire’: el logo y la paleta de colores.

### 6.1. Paleta de colores

La paleta de colores que le dará identidad a StopFire se puede observar en la figura 87.

En un sentido binario, los colores muy cálidos y los fríos permiten al usuario captar el mensaje de un agresor y su opuesto que lo combate, tal como típicamente el agua apaga los incendios. Es una deducción a la que fácilmente puede llegar una persona, con solo ver la representación.

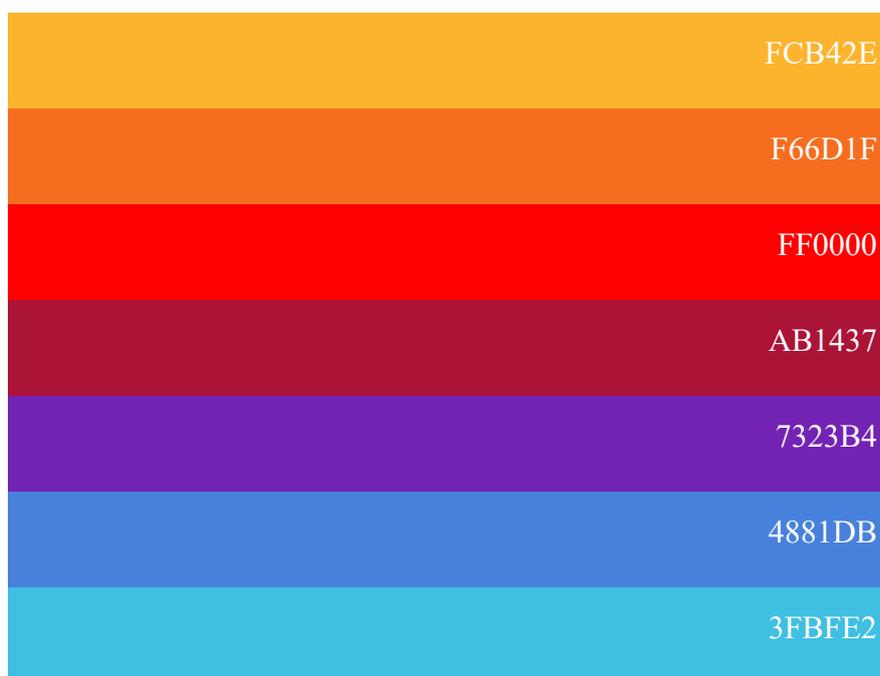


Figura 87: paleta de colores de StopFire

Fuente: elaboración propia

## 6.2. Logo

El logo que se diseñó para StopFire combina dos pictogramas que generan contraposición entre la representación del *fuego* como el objeto al que se apunta erradicar y la *inteligencia computacional* como la herramienta usada para alcanzar el objetivo del sistema.

El mensaje visual se refuerza con los colores vibrantes del primer elemento y la tranquilidad que en oposición se percibe con el azul del agua.

Se utilizó una representación en sentido izquierda-derecha de los pictogramas coloreados, para simular la relación problema-solución, dejándola reposar sobre el nombre de la aplicación con una tipografía bien definida, robusta y que transmite la seguridad de que con esta herramienta se contribuye a combatir una situación que impacta al medio ambiente de forma indiscriminada.



Figura 88: imagotipo de StopFire

Fuente: elaboración propia

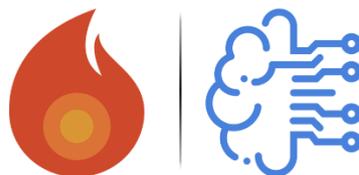


Figura 89: isotipo de StopFire

Fuente: elaboración propia

## 7. Plan de negocio

Esta sección se focaliza en el financiamiento del proyecto.

### 7.1. Costos de desarrollo

Primero, se van a detallar los costos relacionados al desarrollo de la aplicación web y móvil de StopFire (ver tabla 2)

INSUMO	DESCRIPCIÓN	CANTIDAD	COSTO
Arquitectura	Heroku: plataforma de servicios en la nube donde se desplegó la aplicación web (front end)	1	0 USD (uso libre)
	Cloud Provider (AWS): arquitectura donde estará alojado el modelo y el backend	1	200 USD
Recursos humanos	Analista funcional: responsable de realizar tareas de relevamiento, análisis, diseño y documentación	100 hrs	1.500 USD (15 USD / hr)
	Data scientist: responsable de desarrollar el modelo de machine learning	100 hrs	1.800 USD (18 USD / hr)
	Desarrollador full stack: responsable del desarrollo del back-end y front-end de la web y de la aplicación móvil	100 hrs	1.500 USD (15 USD / hr)
	Especialista IoT: encargado de desarrollar el eje de IoT del proyecto	50 hrs	1.000 USD (20 USD / hr)
	Project mánager: líder del proyecto	100 hrs	1.800 USD (18 USD / hr)
<b>PRESUPUESTO TOTAL: 7.800 USD</b>			

Tabla 2: tabla de presupuesto para el desarrollo de StopFire

Fuente: elaboración propia

También hay que tener en cuenta los costos de mantenimiento mensual de la arquitectura en la nube para poder mantener StopFire en funcionamiento.

SERVICIO	DESCRIPCIÓN	COSTO
Amazon EC2	Una instancia Linux t4g.xlarge. vCPUs: 4, Memoria: 16 Gb. Estrategia para colocar precios: Modelo de precios: Instancias bajo demanda. Almacenamiento por instancia: 10 Gb	50.06 USD
Amazon S3	Clases de almacenamiento S3: S3 Standard, Data transfer. Modelo de precios: Precio escalonado para 1 Gb	3 USD
Heroku	Plataforma de servicios en la nube donde se desplegó la aplicación web (front end)	0 USD
<b>PRESUPUESTO MENSUAL DE MANTENIMIENTO: 53.06 USD por mes</b>		

Tabla 3: costo de mantenimiento de la infraestructura de StopFire

Fuente: elaboración propia

Todos estos costos se van a tener en cuenta para poder definir el precio final para los clientes de StopFire en la sección 7.2.

## 7.2. Modelo de negocio

Como se indicó en el estado del arte, el objetivo de este PFI es enfocarse en un desarrollo dirigido netamente a la República Argentina, por lo que StopFire apunta a clientes ubicados en cualquier localidad del país donde haya extensa cubierta forestal (es decir, vegetación vulnerable a incendios).

Algunas de las industrias que se pueden beneficiar por la implementación de StopFire son:

- Sector de agricultura: compuesto principalmente por los campos de cosecha fina (cártamo, cebada, garbanzo y trigo), cosecha gruesa (arroz, girasol y maíz) y de soja. Los campos de sembrado suelen verse amenazados por incendios principalmente durante las épocas del año donde hay sequía y altas temperaturas.
- Sector de ganadería: conformado por la cría y engorde de ganado bovino y porcino. La ignición del combustible (pastizales) no solo afecta al terreno donde crece el ganado, sino que también genera pérdida de animales y de infraestructura en la zona. Adicionalmente, una práctica común en este sector es la quema intencional de pajonales, para ampliar la superficie de pastura. Esto puede causar que se generen incendios que se salgan de control.
- Sector de turismo: los bosques y parques nacionales son lugares propensos a que se inicie un incendio puesto que, si bien, existen lugares habilitados para hacer fuegos, no es una regla que se respeta en todos los casos. Es responsabilidad del Ministerio de Ambiente y Desarrollo Sostenible promover el turismo responsable pero la colaboración de los turistas es indispensable para la prevención de incendios en estas zonas.
- Sector foresto industrial: conformado tanto por bosques nativos como cultivados. Se estima que para el 2016, los incendios rurales afectaron alrededor de 1 millón de hectáreas, de las cuales 247.000 hectáreas eran bosques nativos, 2.500 de bosques cultivados y el resto, pastizales y arbustales (Foresto industria Argentina. 2020).

Considerando los potenciales clientes de StopFire (sector público y privado), se propone un modelo de negocio basado en suscripciones donde el cliente podrá adquirir la solución de StopFire por un precio mensual de 20 o anual de 200 USD.

Los beneficios que obtienen los clientes de StopFire se presentan a continuación:

- Acceso ilimitado a todas las funcionalidades de StopFire
- Actualización de la red neuronal convolucional 1 vez por trimestre (mejoras de la CNN realizada por el equipo de StopFire)

El usuario final de la aplicación móvil va a poder utilizarla de forma gratuita.

## 8. Discusión

La primera dificultad que apareció cuando se empezó con este trabajo de investigación fue poder contactar a un experto en el tema con quien validar el alcance y la utilidad de la solución propuesta. Se logró conseguir al Sr. Martin Quiroga gracias a otro bombero de Mendoza, el Sr. Jorge Herrera, quien fue contactado a través de LinkedIn filtrando por cargo y ubicación.

Luego de concluir la primera etapa de research, se empezó a trabajar en el desarrollo de StopFire. Se decidió priorizar el desarrollo de la red neuronal ya que es el centro del producto de software y sin una CNN que analice imágenes para determinar la existencia del fuego, no existiría producto.

Con respecto al entrenamiento de la CNN, encontrar un data set de uso libre para entrenar y testear la red no fue complicado puesto que los autores de este PFI contaban con experiencia en machine learning y conocían la plataforma Kaggle. Sin embargo, si representó un desafío llegar al modelo final en ConvNeXt.

Primero se probó utilizando ResNet, pero, como se explicó en la sección 3.1 de este PFI, en todos los casos se obtuvo falsos negativos en la etapa de validación, por lo que se descartó esta arquitectura. También se realizaron pruebas usando Transformers, pero el tiempo de inferencia es mayor que el de ConvNeXt y el modelo final era más pesado. Por ello se decidió descartar esos 2 modelos y quedarse con ConvNeXt.

Al momento de desarrollar la aplicación web, se contactaron a expertos en el área para definir cual framework utilizar y se descubrió Streamlit que es un framework basado en Python y, como se cuenta con conocimientos en el equipo en este lenguaje de programación, desarrollar el dashboard en tiempo y forma fue posible.

Sobre la aplicación móvil, se recibió un feedback positivo por parte del bombero que se entrevistó, indicando que el patrullaje voluntario usando una aplicación en un dispositivo móvil *“lo haría un producto innovador, ya que involucrar a la sociedad en la solución los convierte en agentes de cambio”* (ver anexo 11.2).

Se descubrió una infraestructura costosa para desplegar una red de IoT de monitoreo, que en el documento se describe en detalle. Por tal motivo, el MVP implementa el envío de imágenes desde la aplicación y el dispositivo móvil.

## 9. Conclusiones

Luego de realizar distintas iteraciones para desarrollar un modelo de machine learning adecuado para la problemática estudiada en este trabajo de investigación, se logró obtener uno que cumpliera una característica primordial para los autores de este PFI: que no arrojara falsos negativos. Ya que esto implicaría que las columnas de humo no serían detectadas por el modelo y no se generaría la alerta. Permitiendo así la extensión de las llamas en el terreno por lo menos durante 30 minutos más.

Se plantearon también distintos factores para tener en cuenta al momento de implementar una red de dispositivos IoT para la detección y alerta de incendios forestales y se propuso como caso ideal aquella red que logra combinar información capturada por todos los dispositivos existentes: satélites, cámaras y sensores; en contraposición a depender de un solo tipo de dispositivo.

StopFire aporta a la sociedad minimizando el tiempo de reacción ante incendios forestales para así disminuir el daño generado por el fuego en los bosques y campos argentinos.

También facilita el trabajo de las personas encargadas de combatir incendios, ya que entre más temprano se identifique el foco, más rápido se podrán tomar acciones y menor será el impacto en la naturaleza.

Por último, StopFire contribuye en el ahorro de recursos requeridos para combatir el fuego (brigadistas, aviones hidratantes, helicópteros, camionetas, etc.). Puesto que entre más rápido se actúe, menos lograrán expandirse las llamas, por lo que se requerirá de un menor esfuerzo para apagarlas.

Con respecto a la integración de los principales Cloud providers disponibles en el mercado con dispositivos IoT, se analizaron la cantidad de servicios y soluciones de internet de las cosas que cada proveedor ofrece y se llegó a la conclusión que, para el momento en que se está realizando este trabajo de investigación, AWS es el proveedor con más servicios 100% pensados para IoT, es el más popular en el mercado al día de hoy (Hasan. 2022) permitiendo a los dispositivos interactuar con los servicios que ofrece Amazon a través de AWS IoT Core.

StopFire es un trabajo que se integra al proyecto de investigación del INTEC “A21T03 – Aplicaciones de Machine Learning para mejorar el uso de Recursos Naturales”, y es continuación del proyecto final de ingeniería “AQUA: Desarrollo de un modelo de machine learning para prevenir incendios forestales en Pinamar” (Martínez Saucedo. 2021).

También es un proyecto que participó (y fue aceptado) como short paper en la XXVIII edición del Congreso Argentino de Ciencias de la Computación “14260: StopFire: Alertas de Incendios Forestales en Argentina Usando IoT y Machine Learning” (CACIC. 2022), siendo reconocido como un trabajo de investigación relevante a nivel nacional (ver anexo 11.4).

### **9.1. Futuras líneas de investigación**

Como línea de investigación futura, proyectada a desarrollar dentro del INTEC, se propone extender al presente monitoreo a través de captura y análisis de imágenes, con una red de dispositivos IoT conformado por sensores de variables meteorológicas y atmosféricas.

Este tipo de monitoreo por sensores, que incluiría variables climáticas (humedad relativa y temperatura en el ambiente) y del estado del aire (presencia de partículas de humo y de gases combustibles) permitiría obtener indicios de la existencia de incendios en la zona. Agregando así información adicional que sirva para triangular la zona donde se encuentre el potencial foco.

## 10. Bibliografía

Para la redacción de las referencias bibliográficas se utilizó la norma ISO 690, siguiendo las reglas estipuladas por la Facultad de Ingeniería de la UADE.

**ACOT SYSTEM, 2021.** Detección Temprana de Incendios Forestales. [en línea]. [Consulta: 24 abril 2022]. <<https://www.acotsystems.com/deteccion-temprana-incendios-forestales.html>>.

**AMAZON WEB SERVICES, 2022.** Computer Vision Applications at the edge - AWS Panorama - Amazon Web Services. *Amazon Web Services, Inc.* [en línea]. [Consulta: 26 septiembre 2022]. <<https://aws.amazon.com/panorama/>>.

**ARANTEC, 2021.** Sistemas de detección de incendios forestales, tecnología para los bosques. *ARANTEC Smartyplanet · Estaciones de Sensores inalámbricos* [en línea]. [Consulta: 25 septiembre 2022]. <<https://www.arantec.com/sistemas-deteccion-incendios-forestales/>>.

**AWS, 2019.** AWS DeepLens – Cámara de video compatible con el aprendizaje profundo para desarrolladores – AWS. *Amazon Web Services, Inc.* [en línea]. [Consulta: 25 septiembre 2022]. <<https://aws.amazon.com/es/deeplens/>>.

**AWS, 2020.** Solución para productos inteligentes | Implementaciones | Soluciones de AWS. *Amazon Web Services, Inc.* [en línea]. [Consulta: 2 octubre 2022]. <<https://aws.amazon.com/es/solutions/implementations/smart-product-solution/>>.

**BARBA, Salvatore, 2014.** Evaluación ex-ante y ex-post de la precisión de un proyecto fotogramétrico. [en línea]. [Consulta: 27 septiembre 2022]. <[https://www.researchgate.net/figure/Figura-2-Angulo-de-vision-segun-la-distancia-focal\\_fig2\\_293813105](https://www.researchgate.net/figure/Figura-2-Angulo-de-vision-segun-la-distancia-focal_fig2_293813105)>.

**BEDOLLA, Azarely, 2022.** Wildfire Detection Technologies. *Robotics Cats* [en línea]. [Consulta: 27 septiembre 2022]. <<https://roboticscats.com/2022/07/27/wildfire-detection-technologies/>>.

**BOSCH, 2021.** Cámara de detección de incendios y humo AVIOTEC. *Bosch Security and Safety Systems I Latin America* [en línea]. [Consulta: 15 mayo 2022]. <<https://www.boschsecurity.com/xl/es/soluciones/sistemas-de-deteccion-de-incendios/camara-de-deteccion-de-incendios-y-humo/>>.

**CACIC, 2022.** Short papers aceptados. [en línea]. [Consulta: 20 octubre 2022]. <<https://cacic2022.unlar.edu.ar/index.php/acceptedshort/>>.

**CAMSAT, 2020.** All-year solar power system for CCTV cameras - iCAM-Solar365 P30W. *Camsat* [en línea]. [Consulta: 24 septiembre 2022]. <<https://camsat-cctv.com/icam-solar365-p30w/>>.

- 
- CRUZ, Henry, ECKERT, Martina, MENESES, Juan y MARTÍNEZ, José-Fernán, 2016.** Efficient Forest Fire Detection Index for Application in Unmanned Aerial Systems (UASs). *Sensors*, vol. 16, no. 6, pp. 893. ISSN 1424-8220. DOI 10.3390/s16060893.
- CUI, Pinchen, 2017.** Comparison of IoT Application Layer Protocols. En: Accepted: 2017-04-25T22:05:09Z [en línea], [Consulta: 7 junio 2022]. <<https://etd.auburn.edu/handle/10415/5713>>.
- DINCER, Baris, 2021.** Wildfire Detection Image Data. [en línea]. [Consulta: 24 abril 2022]. <<https://www.kaggle.com/brsdincer/wildfire-detection-image-data>>.
- DOSOVITSKIY, Alexey, BEYER, Lucas, KOLESNIKOV, Alexander, WEISSENBORN, Dirk, ZHAI, Xiaohua, UNTERTHINER, Thomas, DEGHANI, Mostafa, MINDERER, Matthias, HEIGOLD, Georg, GELLY, Sylvain, USZKOREIT, Jakob y HOULSBY, Neil, 2021.** An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. En: arXiv:2010.11929 [cs]version: 2type: article [en línea]. S.l.: arXiv. [Consulta: 9 junio 2022]. arXiv:2010.11929. <<http://arxiv.org/abs/2010.11929>>.
- EARTH SCIENCE DATA SYSTEMS, NASA, 2017.** What is Data Latency? *Earthdata* [en línea]. [Consulta: 24 septiembre 2022]. <<http://www.earthdata.nasa.gov/learn/backgrounders/data-latency>>.
- EARTH SCIENCE DATA SYSTEMS, NASA, 2021.** About FIRMS. *Earthdata* [en línea]. [Consulta: 24 septiembre 2022]. <<http://www.earthdata.nasa.gov/learn/find-data/near-real-time/firms/about-firms>>.
- EDPRICE-MSFT, 2022.** Condition monitoring for industrial IoT - Azure Architecture Center. [en línea]. [Consulta: 2 octubre 2022]. <<https://learn.microsoft.com/en-us/azure/architecture/solution-ideas/articles/condition-monitoring>>.
- FAST AI, 2022.** Data augmentation in computer vision | fastai. [en línea]. [Consulta: 23 julio 2022]. <[https://docs.fast.ai/vision.augment.html#aug\\_transforms](https://docs.fast.ai/vision.augment.html#aug_transforms)>.
- FORESTO INDUSTRIA ARGENTINA, 2020.** Manejo del Fuego - Foresto Industria. [en línea]. [Consulta: 20 octubre 2022]. <<https://forestoindustria.org.ar/manejo-del-fuego/>>.
- GHOSH, Anirudha, SUFIAN, A., SULTANA, Farhana, CHAKRABARTI, Amlan y DE, Debashis, 2020.** Fundamental Concepts of Convolutional Neural Network. . S.l.: s.n., pp. 519-567. ISBN 978-3-030-32643-2.
- GODDARD MEDIA STUDIOS, NASA's, 2022.** GMS: GOES Satellites Wildfire Detection and Monitoring. [en línea]. [Consulta: 24 septiembre 2022]. <<https://svs.gsfc.nasa.gov/14108>>.
- GOOGLE CLOUD, 2022.** Cloud IoT Core. *Google Cloud* [en línea]. [Consulta: 2 octubre 2022]. <<https://cloud.google.com/iot-core?hl=es>>.
- GROSS, Bernd, 2022.** Google is sunsetting IoT Core. Now what? *Software AG* [en línea]. [Consulta: 2 octubre 2022]. <<https://blog.softwareag.com/iot-core/>>.
- HASAN, Mohammad, 2022.** The IoT cloud: Microsoft Azure vs. AWS vs. Google Cloud. *IoT Analytics* [en línea]. [Consulta: 7 junio 2022]. <<https://iot-analytics.com/iot-cloud/>>.
-

- 
- HERNÁNDEZ-HOSTALLER, Néstor, 2017.** Evaluación de tecnologías de sensores para la detección temprana de incendios forestales | Revista Tecnología en Marcha. [en línea], [Consulta: 21 septiembre 2022]. <[https://revistas.tec.ac.cr/index.php/tec\\_marcha/article/view/3043](https://revistas.tec.ac.cr/index.php/tec_marcha/article/view/3043)>.
- HIKVISION, 2019.** Solar-Powered Standalone Security Camera. *hiknow* [en línea]. [Consulta: 24 septiembre 2022]. <<https://www.hikvision.com/es/products/IP-Products/Network-Cameras/solar-powered-standalone-security-camera/>>.
- HIKVISION, 2020.** DS-2CD2T87G2-LSU/SL. *hiknow* [en línea]. [Consulta: 27 septiembre 2022]. <<https://www.hikvision.com/es-la/products/IP-Products/Network-Cameras/Pro-Series-EasyIP/-ds-2cd2t87g2-lsu-sl/>>.
- HIKVISION, 2022.** Productos térmicos. *hiknow* [en línea]. [Consulta: 15 mayo 2022]. <<https://www.hikvision.com/es-la/products/Thermal-Products/>>.
- How to Get Started with AWS IoT Core Quick Connect* [en línea], 2020.** [Consulta: 7 junio 2022]. <[https://www.youtube.com/watch?v=6w9a6y\\_-T2o](https://www.youtube.com/watch?v=6w9a6y_-T2o)>.
- HUAWEI, 2018.** Huawei Power Solutions. *Huawei Power Solutions* [en línea]. [Consulta: 25 septiembre 2022]. <<https://www.huaweipowersolutions.com/>>.
- IMPERVA, 2021.** What is OSI Model. *Learning Center* [en línea]. [Consulta: 7 junio 2022]. <<https://www.imperva.com/learn/application-security/osi-model/>>.
- INCHAUSTI, Pablo Ezequiel, MARTINEZ SAUCEDO, Ana Carolina, AMET, Leonardo Javier, BLANCO, Pedro Martin, NIEVAS, Guillermo Manuel y GIUSTO, Luana, 2022.** Aplicaciones de Machine Learning para el uso Sustentable de Recursos Naturales. *Workshop de Investigadores en Ciencias de la Computación. Edición 2022 - Universidad Champagnat, Mendoza, Argentina.* [en línea]. S.l.: s.n., [Consulta: 29 abril 2022]. <<https://wicc2022.tk/workshop/6256d0d67c76870009464c77/post/6260a4135fedd10097c5c73>>.
- KURTZ, Dirmar Bernardo, PERUCCA, Alba Ruth y SAUCEDO, Griselda, 2022.** Al 21 de febrero de 2022, la superficie quemada fue de 934.238 hectáreas | Instituto Nacional de Tecnología Agropecuaria. [en línea]. [Consulta: 24 abril 2022]. <<https://inta.gob.ar/noticias/al-21-de-febrero-de-2022-la-superficie-quemada-fue-de-934238-hectareas>>.
- MARTÍNEZ SAUCEDO, Ana Carolina, 2021.** *AQUA: Desarrollo de un Modelo de Machine Learning para Prevenir Incendios Forestales en Pinamar.* Tesis de Ingeniería en Informática inédita. S.l.: Universidad Argentina de la Empresa (UADE).
- MATICH, Damián Jorge, 2001.** *Redes Neuronales: Conceptos Básicos y Aplicaciones* [en línea]. 1 marzo 2001. S.l.: s.n. <[https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5\\_anio/orientadora1/monografias/matich-redesneuronales.pdf](https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monografias/matich-redesneuronales.pdf)>.
-

---

**MICROSOFT, 2021.** Microsoft Azure IoT Reference Architecture. [en línea]. [Consulta: 7 junio 2022]. <<https://azure.microsoft.com/en-us/resources/microsoft-azure-iot-reference-architecture/>>.

**MINISTERIO DE AMBIENTE Y DESARROLLO SOSTENIBLE, 2021.** Manejo del Fuego. Reporte de incendios. *Argentina.gob.ar* [en línea]. [Consulta: 15 abril 2022]. <<https://www.argentina.gob.ar/ambiente/fuego/diciembre-2021>>.

**ORACLE, 2019.** ¿Qué es el Internet de las cosas (IoT)? [en línea]. [Consulta: 15 mayo 2022]. <<https://www.oracle.com/ar/internet-of-things/what-is-iot/>>.

**PANO AI, 2022.** Pano AI. [en línea]. [Consulta: 25 septiembre 2022]. <<https://www.pano.ai/>>.

**PLATNICK, Steve, 2017.** The Earth Observer: Mar - Apr, 2017. [en línea]. [Consulta: 24 septiembre 2022]. <<https://eosps.nasa.gov/earthobserver/mar-apr-2017>>.

**PROMAX, 2019.** Cámaras Termográficas: Cómo funcionan y por qué pueden ser necesarias. [en línea]. [Consulta: 25 septiembre 2022]. <<https://www.promax.es/esp/noticias/400/camara-termografica-como-funcionan-y-por-que-pueden-ser-necesarias/>>.

**RABE, Tom, 2021.** Robots, drones and satellites: NSW to research new bushfire fighting technology. [en línea]. [Consulta: 21 septiembre 2022]. <<https://www.smh.com.au/national/nsw/robots-drones-and-satellites-nsw-to-research-new-bushfire-fighting-technology-20210616-p581li.html>>.

**SCHULKIN, Julieta, 2020.** La NASA seleccionó un proyecto de jóvenes argentinos que busca controlar incendios forestales con tecnología. *infobae* [en línea]. [Consulta: 15 abril 2022]. <<https://www.infobae.com/tecno/2020/10/10/la-nasa-selecciono-un-proyecto-de-jovenes-argentinos-que-busca-controlar-incendios-forestales-con-tecnologia/>>.

**SERVICIO METEOROLÓGICO NACIONAL, 2022.** Servicio Meteorologico Nacional. [en línea]. [Consulta: 26 septiembre 2022]. <[https://www.smn.gob.ar/indices\\_peligro\\_fuego](https://www.smn.gob.ar/indices_peligro_fuego)>.

**SERVICIO NACIONAL DE MANEJO DEL FUEGO, 2018.** ¿Cuáles son las variables y qué factores las afectan? *Argentina.gob.ar* [en línea]. [Consulta: 11 septiembre 2022]. <<https://www.argentina.gob.ar/ambiente/fuego/conocemas/variables>>.

**SERVICIO NACIONAL DE MANEJO DEL FUEGO, 2021.** Diciembre de 2021. *Argentina.gob.ar* [en línea]. [Consulta: 23 octubre 2022]. <<https://www.argentina.gob.ar/ambiente/fuego/diciembre-2021>>.

**SIGNALS IOT, 2020.** Los diez protocolos más importantes utilizados para IoT. *Signals IoT* [en línea]. [Consulta: 2 octubre 2022]. <<https://signalsiot.com/los-diez-protocolos-mas-importantes-utilizados-para-iot/>>.

**SOLER, Paula, 2021.** Incendios forestales: a los 16 años, crearon una app para detectar los focos de manera temprana. *LA NACION* [en línea]. [Consulta: 15 abril 2022]. <<https://www.lanacion.com.ar/sociedad/incendios-forestales-a-los-16-anos-crearon-una-app-para-detectar-los-focos-de-manera-temprana-nid23032021/>>.

- 
- STRICKLAND, Ashley, 2022.** Este nuevo satélite meteorológico podría detectar incendios forestales antes que nosotros. *CNN* [en línea]. [Consulta: 24 septiembre 2022]. <<https://cnnespanol.cnn.com/2022/03/02/nuevo-satelite-meteorologico-goes-18-incendios-forestales-trax/>>.
- SZELISKI, Richard, 2011.** *Computer Vision. Algorithms and Applications*. Springer. S.l.: s.n. ISBN 978-1-84882-934-3.
- TELEDYNE FLIR, 2022.** Edge Computing. [en línea]. [Consulta: 26 septiembre 2022]. <<https://www.flir.com/discover/iis/machine-vision/edge-computing/>>.
- THOMAS, Gregory, 2021.** Startup Uses AI, Panoramic Cameras for Wildfire Detection. *GovTech* [en línea]. [Consulta: 25 septiembre 2022]. <<https://www.govtech.com/public-safety/startup-uses-ai-panoramic-cameras-for-wildfire-detection>>.
- TOPOGRAPHIC-MAP, 2022.** Mapa topográfico Argentina, altitud, relieve. *topographic-map.com* [en línea]. [Consulta: 26 septiembre 2022]. <<https://es-ar.topographic-map.com/maps/6nhh/Argentina/>>.
- YEUNG, Tiffany, 2022.** What's the Difference: Edge Computing vs Cloud Computing. *NVIDIA Blog* [en línea]. [Consulta: 2 octubre 2022]. <<https://blogs.nvidia.com/blog/2022/01/05/difference-between-cloud-and-edge-computing/>>.

## 11. Anexos

### 11.1. Glosario

<b>Término</b>	<b>Definición</b>
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface. Es un conjunto de requerimientos que permite la comunicación de datos entre aplicaciones
API REST	Es una API que se ajusta a los límites de la arquitectura REST y facilita la comunicación de un sitio web con otras aplicaciones
AWS	Amazon Web Services. Es un proveedor de computación en la nube
Cámara estanca	Es un dispositivo diseñado para funcionar en sus condiciones de estanqueidad iniciales
CoAP	Constrained Application Protocol
Dashboard	Panel de control. Es una herramienta de gestión de la información que sirve para monitorear y analizar y muestra de manera visual los indicadores relevantes para un negocio
Datasets	Conjunto de datos que se usan para entrenar, probar y validar un modelo de machine learning
Deep learning	Es una rama de la inteligencia artificial cuyo objetivo es hacer que las computadoras imiten el comportamiento humano para poder realizar tareas que solamente pueden ser realizadas por una persona
Edge computing	Computación en el borde, en inglés
Epoch	Es un ciclo completo a través de todo el conjunto de datos de entrenamiento de una red neuronal
FastAI	Es una librería de Deep learning basada en PyTorch que facilita el la creación y entrenamiento de redes neuronales

IDE	Integrated Development Environment
Imagotipo	Es el resultado de la unión del logotipo y el isotipo. La parte tipográfica puede ser separada de la parte icónica. Un ejemplo es el imagotipo de 'Amazon'
INTA	Instituto Nacional de Tecnología Agropecuaria
ISM	Industrial, Scientific, and Medical
Logotipo	Es una palabra escrita de una forma que permite que sea memorable. Un ejemplo es el logotipo de 'Coca-Cola'
LPWAN	Low-Power Wide-Area Network
Machine learning	Aprendizaje automatizado, en inglés
Matriz de confusión	Matriz de 2x2 que muestra la cantidad de verdaderos positivos y negativos, junto a los falsos positivos y negativos
MQTT	Message Queueing Telemetry Transport
MVP	Minimum Viable Product. Busca demostrar con un costo relativamente bajo si un producto de software es un potencial éxito o no. Optimizando así la inversión en tiempo y dinero.
Overfitting	En castellano: sobre ajuste, sobre entrenamiento o sobre aprendizaje. Ocurre cuando se genera un modelo con un accuracy alto (e.g.: > 90%) en la etapa de entrenamiento, pero con un bajo accuracy en la etapa de testeo (e.g.: < 60%). Es decir, se obtiene un modelo que no es generalizable para nuevos datos
Pajonales	Maleza que no sirve como alimento para los animales
PDU	Protocol Data Units
PFI	Proyecto Final de Ingeniería
Protección IP65	Los dispositivos con clase de protección IP65 están protegidos contra la entrada de polvo y el contacto. La carcasa protege el interior del equipo contra los chorros de agua desde cualquier ángulo
SDK	Software Development Kit
SNMF	Servicio Nacional de Manejo del Fuego

## 11.2. Entrevistas realizadas

Fecha: 17/05/2022

Entrevistado: Martin Quiroga

Cargo: Jefe de brigada y responsable del departamento de incendios forestales de la federación Mendocina

Contacto: bvlc13@yahoo.com.ar

### **¿Cuál es su cargo y desde hace cuánto tiempo ejerce como bombero?**

El Sr. Martín es jefe la brigada y del departamento de incendios forestales de la provincia de Mendoza desde hace 18 años. También trabaja como bombero voluntario desde hace 23 años para la federación de bomberos voluntarios.

### **¿Cómo se detectan los incendios forestales en la Argentina? Lo hacen a través de llamadas solamente o utilizan algún software para analizar variables para predecir potenciales zonas de incendios (como humedad, temperatura, viento, etc.)**

Ellos a nivel federativo de los bomberos voluntarios no tienen ningún tipo de detección excepto que venga un pedido oficial de Defensa Civil a través de los organismos naturales. Es decir, desde los bomberos de la policía o a través de un llamado directo en caso de que haya un evento grande.

Por lo general, la brigada de bomberos voluntarios trabaja fuera de la provincia o directamente en sus jurisdicciones como bomberos de su cuartel.

Por otro lado, en el servicio Provincial del Manejo del fuego sí utilizan páginas de uso gratuito y público para detectarlos.

Una de las que más utilizas es FireMaps porque es de las más confiables gracias al tipo de barrido que hacen los satélites y porque pasan información en tiempo real de los puntos de calor detectados. Para detectar los incendios utilizan pixeles que marcan el calor. Si con un barrido satelital se detecta un punto de calor de 1 o 2 pixeles y al siguiente barrido son 4 pixeles, esta es una clara señal de que es un incendio que se les escapó.

---

**¿Cuántos grupos trabajan para combatir un incendio y cuál es el proceso para comunicarse entre sí?**

Cuando se recibe el llamado, sale la cuadrilla y si hay que pedir apoyo, viene una de las cuadrillas que está de franco para ayudar y así sucesivamente hasta poder combatirlo.

**¿Cuánto tiempo tarda un incendio en propagarse a un punto que se vuelve difícil de controlar?**

La respuesta va a depender mucho del tipo de combustible que tenga, si tiene continuidad vertical/horizontal, si hay inclinación en el suelo, si existen condiciones de arder, etc.

Que se den las condiciones, o no, va a variar según el porcentaje de humedad que haya, la velocidad del viento, la inclinación del terreno que ayude a la propagación.

Asumiendo que hay condiciones para propagarse, si el incendio se atiende en las primeras 2 horas, sería lo ideal.

**Después, se le explicó que nuestra propuesta de tiempo para poder mandar las fotos era de 30 minutos y se le preguntó si le parecía bien. Su respuesta a continuación**

Lo antes posible es lo ideal. Mientras más rápido la cámara detecte columnas de humo, es mejor.

Asumiendo que se instalan cámaras en puntos estratégicos, si las imágenes son analizadas cada 30 minutos es ideal porque es como tener a un vigía en una parada en altura. Esto es lo que se hace en algunos parques nacionales para poder monitorear la zona constantemente.

**¿Cuántas falsas alarmas de incendios reciben por mes/año?**

Si es común. No sabe cuánto es el porcentaje de falsas alarmas vs. el total de llamados que reciben al mes, pero si sabe que es un número significativo.

También hay muchas denuncias que no se hacen porque está muy arraigado en la sociedad quemar cosas y no toman en cuenta otros aspectos como la conservación del ambiente

---

Incluso en la aplicación que ellos utilizan puede pasar que se detecten puntos de calor que no son incendios forestales (como basurales, por ejemplo), sobre los que ellos no tendrían que actuar a menos que se salgan de control.

Fecha: 20/09/2022

Entrevistado: Martin Quiroga

Cargo: Jefe de brigada y responsable del departamento de incendios forestales de la federación Mendocina

Contacto: bvlc13@yahoo.com.ar

**Para este PFI estamos proponiendo, aparte de la aplicación para computadora donde se puedan recibir la alerta de incendios, crear una aplicación para teléfonos para que los transeúntes que vean un incendio en la zona puedan tomar una foto y enviarla para alertar. El envío de la foto va a servir para tomar la geolocalización de la persona que hizo la alerta y que los bomberos puedan ir exactamente al lugar de los hechos. Quisiéramos crear una especie de red de “patrullaje voluntario” que contribuya al monitoreo y la prevención de incendios forestales. ¿Qué le parece esa idea? ¿Desde su experiencia, cree que tiene sentido la propuesta? ¿Es algo que ya existe en el país, pero yo no estoy al tanto?**

Desde mi conocimiento, en países de Europa, Australia, Canadá, o Estados Unidos posiblemente si existe una aplicación así, pero en la Argentina no.

Me parece que actualmente no es una aplicación que esté al alcance de cualquier persona y creo que eso es lo que lo haría un producto innovador, ya que involucrar a la sociedad en la solución los convierte en agentes de cambio.

**Otra funcionalidad que se nos ocurrió para la aplicación de teléfonos es que sirva para alertar a campistas y personas que les guste estar en bosques al momento de su acampado. Por ejemplo, si yo voy a ir a un bosque, puedo marcar mi ubicación y, en caso de que haya una alerta de incendios a X kilómetros a la redonda de mi ubicación, la aplicación podría mandarme una alerta. Ahora, nuestra pregunta para usted es: ¿cuál sería la distancia de km a la redonda que sugieren los bomberos para que la gente pueda irse de una vez porque el incendio puede llegar a donde estén acampando?**

Esta es una pregunta completa a que la respuesta va a variar en función de múltiples factores: inclinación del terreno, insolación del combustible, viento, ubicación de las personas, etc.

Tomando esos tres parámetros, en el peor escenario, la distancia ideal a la que se le debería alertar a una persona que debe evacuar la zona inmediatamente va a ser alrededor de 5 km a la redonda.

De vuelta, va a depender mucho de las 3 variables así que no se puede tomar por seguro, pero es un buen estimado.

**Y, por último, ¿tiene algún contacto que trabaje en parques nacionales que nos pueda decir cuántos parques/bosques argentinos hoy en día tienen vigilancia 100% por cámara, cuantos dependen al 100% de los vigías y en cuantos hay una combinación (es decir, que usen cámaras y también vigías ubicados en altura)? Nos gustaría entender si tiene sentido proponer “reusar” la infraestructura de cámaras que ya existe en el país para el proyecto.**

No tengo ese dato, pero puedo averiguarlo a través de técnicos que trabajan en el servicio nacional.

Algo para tener en cuenta es que el gobierno no quiere invertir en cámaras porque como la web FireMap es gratis, prefieren que se utilice para monitorear los incendios en vez de invertir en infraestructura propia.

Sé que, en EE. UU., Canadá y otros países más avanzados se invierte recurrentemente en infraestructura local, pero en el territorio nacional no algo que está disponible en todos los parques.

### 11.3. Tabla de figuras

Figura 1: representación del triángulo del comportamiento del fuego.....	13
Figura 2: estructura de una red neuronal totalmente conectada .....	16
Figura 3: modelo básico conceptual de una CNN .....	19
Figura 4: arquitectura de una CNN .....	19
Figura 5: matriz de confusión de un modelo de clasificación binario.....	20
Figura 6: servicios IoT en la nube por proveedor.....	24
Figura 7: capas del modelo OSI .....	25
Figura 8: arquitectura de MQTT .....	27
Figura 9: arquitectura IoT simple conectada a AWS .....	27
Figura 10: arquitectura de AMQP .....	28
Figura 11: conectividad de dispositivos con Azure IoT Hub.....	28
Figura 12: comunicación CoAP .....	29
Figura 13: descripción general del modelo .....	31
Figura 14: comparación con el estado del arte en benchmarks de clasificación de imágenes .	32
Figura 15: imágenes modelos de incendios forestales .....	34
Figura 16: ejemplo de transformaciones usando “aug_transforms” de FastAI.....	35
Figura 17: imágenes listas para ser usadas en el entrenamiento de la red neuronal.....	35
Figura 18: learning rate Finder obtenido en el entrenamiento de la red ConvNeXt .....	37
Figura 19: matriz de confusión del modelo ResNet34 .....	38
Figura 20: falsos positivos y negativos .....	39
Figura 21: matriz de confusión del modelo ResNet50 .....	40
Figura 22: falsos positivos y negativos .....	40
Figura 23: matriz de confusión del modelo ResNet101 .....	41
Figura 24: matriz de confusión del modelo ResNet101 to _fp16.....	42
Figura 25: falsos negativos.....	42
Figura 26: matriz de confusión del modelo ConvNeXt.....	44
Figura 27: página principal – Mapa de cámaras.....	45
Figura 28: módulo de incendios – Reporte de incendios .....	46
Figura 29: página institucional con información del proyecto.....	47

---

Figura 30: wireframes. Landing page.....	48
Figura 31: wireframes. Pantalla de inicio.....	49
Figura 32: wireframes. Capturar foto de incendio.....	49
Figura 33: wireframes. Mensaje de confirmación de alerta .....	50
Figura 34: wireframes. Mensaje de error .....	51
Figura 35: arquitectura de StopFire.....	52
Figura 36: diagrama de base de datos de StopFire .....	53
Figura 37: mapa de índices de peligro de incendios en Argentina.....	54
Figura 38: mapa topográfico de Argentina.....	55
Figura 39: flota de observación de la Tierra de la NASA donde se aprecian los 4 satélites que alimentan FIRMS .....	57
Figura 40: puntos de calor detectados .....	57
Figura 41: representación artística de una nave espacial GOES en órbita alrededor de la Tierra .....	58
Figura 42: ejemplo de cámara termográfica.....	60
Figura 43: AWS DeepLens .....	61
Figura 44: instalación de la estación Pano en Healdsburg, California.....	62
Figura 45: detección de columnas de humo usando el sistema Pano.....	62
Figura 46: detección fotoeléctrica de humo .....	63
Figura 47: solución de ARANTEC para la detección de incendios forestales en Extremadura .....	64
Figura 48: métodos y variables utilizadas para la monitorización del fuego en el bosque .....	65
Figura 49: cámara solar – Hikvision .....	67
Figura 50: Power Cube – Huawei .....	67
Figura 51: iCAM-Solar365 – CamSAT .....	68
Figura 52: representación de edge computing vs cloud computing .....	71
Figura 53: cámara Hikvision DS-2CD2T87G2-LSU/SL .....	73
Figura 54: representación del ángulo de visión según la distancia focal .....	74
Figura 55: AWS DeepLens (edición 2019).....	75
Figura 56: flujo de AWS Panorama .....	76
Figura 57: ejemplo de arquitectura AWS para productos inteligentes.....	78

---

---

Figura 58: ejemplo de arquitectura unificada de comunicación de dispositivos IoT con servicios de Azure .....	79
Figura 59: ejemplo de arquitectura en Google Cloud .....	80
Figura 60: diagrama con tecnologías comunes de detección de incendios forestales .....	81
Figura 61: análisis de radio enlace (Formosa, Argentina) usando Ubiquiti ISP Design Center .....	82
Figura 62: modelo del framework ágil utilizado en este PFI .....	83
Figura 63: imagen #1 con neblina y resultados .....	86
Figura 64: imagen #2 con neblina y resultados .....	86
Figura 65: imagen #1 de tormenta de arena con resultados .....	87
Figura 66: imagen #2 de tormenta de arena con resultados .....	87
Figura 67: imagen #3 de tormenta de arena con resultados .....	87
Figura 68: imagen de pastizal sin fuego con resultados .....	88
Figura 69: imagen de pastizal en llamas con resultado .....	88
Figura 70: imagen de humo blanco con resultados .....	89
Figura 71: imagen de humo negro con resultados.....	89
Figura 72: imagen con leve presencia de llamas .....	89
Figura 73: imagen con presencia media de llamas .....	90
Figura 74: imagen con alta presencia de llamas .....	90
Figura 75: imagen #1 de incendio y humo desde una vista contrapicada con resultados .....	90
Figura 76: imagen #2 de incendio y humo desde una vista contrapicada con resultados .....	91
Figura 77: landing page de la aplicación móvil.....	91
Figura 78: pantalla de inicio con el botón de “iniciar cámara” para realizar la alerta.....	92
Figura 79: cámara activada capturando una imagen desde la computadora para simular el incendio visto desde el celular.....	92
Figura 80: foto capturada desde la aplicación móvil.....	93
Figura 81: pantalla de confirmación una vez se envía la foto seleccionando el botón de “Enviar” en la pantalla anterior .....	93
Figura 82: POST recibido en el backend.....	94
Figura 83: base de datos actualizada con la alerta.....	94
Figura 84: dashboard con las alertas desde el 25 de julio de 2022 al 23 de octubre del 2022 .	94

---

Figura 85: mapa de incendios con las alertas desde el 25 de julio de 2022 al 23 de octubre del 2022 .....	95
Figura 86: mensaje de error por fallo en el envío de la alerta .....	95
Figura 87: paleta de colores de StopFire .....	96
Figura 88: imagotipo de StopFire.....	97
Figura 89: isotipo de StopFire .....	97
Figura 90: cronograma final .....	124

## 11.4. Paper aceptado en CACIC 2022

En esta sección se expone el short paper presentado en la XXVIII edición del Congreso Argentino de Ciencias de la Computación (CACIC 2022).

### StopFire: Alertas de Incendios Forestales en Argentina Usando IoT y Machine Learning

Alejandra Curbelo<sup>1</sup> [0000-0003-1708-6624], Juan Cruz Alric<sup>1</sup> [0000-0002-8854-9844] y Pablo Ezequiel Inchausti<sup>1</sup> [0000-0002-8342-1796]

<sup>1</sup>Universidad Argentina de la Empresa (UADE), Instituto de Tecnología (INTEC). Buenos Aires, Argentina  
{alcurbelo, jalric, pinchausti}@uade.edu.ar

**Resumen.** Los incendios forestales son devastadores para un ambiente y sus efectos se miden en cientos de miles de hectáreas. Para contribuir en la prevención de incendios forestales, se desarrolla una solución de monitoreo por imágenes de las zonas en riesgo basada en IoT, y la utilización de modelos de Machine Learning con Redes Neuronales Convolucionales para identificar en las imágenes la presencia del fuego. La solución se completa con un tablero de monitoreo de los dispositivos de IoT y una aplicación móvil para reportar el riesgo de incendio enviando imágenes geolocalizadas.

**Keywords:** Incendios Forestales. Machine Learning, IoT, Redes Neuronales

## 1 Introducción

Un incendio forestal se define como un fuego que se propaga rápidamente y de forma descontrolada a través de la vegetación de una zona, y, si bien se pueden originar por causas naturales, como las tormentas eléctricas, más del 95% de los incendios forestales en Argentina son provocados por el hombre, ya sea de forma intencional, o no [1].

Los incendios forestales suelen producir efectos devastadores, y de acuerdo al Servicio Nacional de Manejo del Fuego (SNMF), en el 2021 más de 302 mil hectáreas en Argentina fueron afectadas por el fuego, con la provincia de San Luis encabezando la lista [2]. En febrero del 2022, los incendios forestales de Corrientes afectaron más de 900 mil hectáreas, un equivalente al 11% de la superficie provincial [3].

El objetivo del presente trabajo es contribuir a la prevención de incendios forestales en Argentina, y desarrollar un sistema de alertas y monitoreo por imágenes. Se utiliza IoT para el monitoreo y se determina la presencia del fuego con modelos de Machine Learning (ML). Se complementa la solución con una aplicación móvil para reportar el riesgo de incendio en una zona, enviando imágenes con información geolocalizada.

El trabajo inicia en el 2022 como Proyecto Final de Ingeniería Informática (PFI) de Alejandra Curbelo y Juan Cruz Alric, siendo Pablo Inchausti, docente en PFI, el tutor.

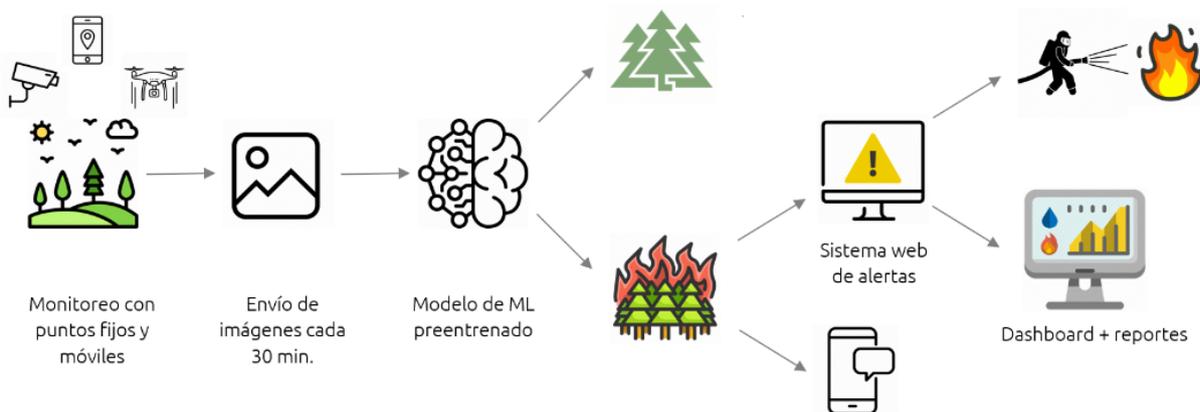
También se integra en el INTEC, el Instituto de Tecnología de UADE, a la línea A21T03: *Aplicaciones de Machine Learning para el uso de Recursos Naturales*. [4] y continúa el trabajo de *AQUA* [5] sobre prevención de incendios forestales en Pinamar.

## 2 Materiales y Métodos

El sistema StopFire está compuesto por una aplicación web, con un módulo de detección de incendios forestales que integra un modelo de Machine Learning para analizar imágenes de zonas forestales capturadas por dispositivos de IoT.

Los dispositivos de IoT se conectan a servicios desplegados en el Cloud Provider AWS, y reciben imágenes cada 30 minutos, para procesarlas y generar una alerta en caso de que el modelo entrenado detecte que existe un posible incendio en la zona.

El sistema de monitoreo se presenta como un tablero que permite ver las cámaras conectadas a la red, con su ubicación geográfica. En el módulo de alertas se muestra la foto que el modelo utilizó para determinar el potencial foco de incendio, con información geolocalizada de la cámara y el día y la hora de la captura de la imagen. En la **Fig. 1.** se describe el modelo de solución de StopFire a alto nivel:



**Fig. 1.** StopFire: modelo de solución

Respecto a la aplicación móvil, permitiría que, de forma voluntaria, las personas presentes en zonas forestales puedan reportar incidentes con riesgo de incendios. Desde la aplicación se permite el envío de imágenes al sistema de monitoreo y analizarlas con los modelos de detección de incendios, y también obtener la información geolocalizada del lugar y el tiempo de la captura de la imagen para identificar la zona de riesgo.

Teniendo en cuenta que el factor humano es responsable de más del 95% de los incendios forestales [1], la aplicación móvil permite involucrar de forma directa a nuestra sociedad, y le permite contribuir en la prevención reportando incidentes desde la zona de riesgo, y así reducir el nivel de responsabilidad en estos desastres naturales.

De esta forma, también el envío de imágenes mediante las aplicaciones móviles permite escalar la red de monitoreo más allá de los dispositivos de IoT desplegados en la zona. Y el análisis de las imágenes enviadas, también sirven tanto para mejorar el entrenamiento de los modelos, como para identificar lugares propensos a incendios con el objetivo de ampliar la red de dispositivos.

La tecnología para el desarrollo de los modelos de Machine Learning (ML) está basada en Redes Neuronales Artificiales (RNA), técnicas de visión por computadora y Redes Neuronales Convolucionales (CNN) que son especialmente efectivas en campos como la visión artificial.

### 2.1 Redes Neuronales Convolucionales (CNN)

La Red Neuronal Convolutiva (CNN) es un tipo de red neuronal multicapa o arquitectura de aprendizaje profundo inspirada en el sistema visual de los seres vivos. Tienen una arquitectura *feed forward*, que significa que las neuronas de cada capa se conectan con todas las neuronas de la capa siguiente, pero no con neuronas de la misma capa. Las primeras capas de la red aprenden y extraen las características de alto nivel, mientras que las capas más profundas, extraen las características de bajo nivel. [6]

En las CNN, hay tres tipos de capas intermedias: la capa convolutiva, que es el componente más importante de las arquitecturas CNN porque es la que genera un mapa con las características de las imágenes a partir de las imágenes de entrada. La capa de agrupación, que toma las características de mayor tamaño y las reduce en mapas de características de menor tamaño. Y finalmente, la capa completamente conectada, que se utiliza como clasificador, y que, a partir de la última capa de agrupación, se pasa por la red *feed forward* para generar la salida. [6]

Para el desarrollo de la CNN se utilizó la librería *fast.ai* [7], que se especializa en Deep Learning, para creación de las redes neuronales.

### 2.2 Preparación de los datos

Para el entrenamiento de la red neuronal, se usaron conjuntos de datos públicos de incendios forestales provenientes de plataformas libres como Kaggle. Por ejemplo, *Wildfire Detection Image Data* [8] es un conjunto de datos con imágenes de uso libre de incendios forestales para entrenar modelos de Machine Learning y Deep Learning.

El procedimiento inicia con la carga de las 1.832 imágenes etiquetadas de Kaggle, y adicionalmente, para aumentar la muestra disponible para el entrenamiento, se utilizó la librería *fast.ai* y su función *aug\_transforms* para crear una lista de transformaciones a partir de rotaciones, acercamientos, deformaciones e iluminación sobre los datos de entrenamiento. Adicionalmente se utilizaron imágenes de incendios reales en Argentina para probar el rendimiento del modelo con datos locales de Argentina.

En la **Fig. 2** se pueden ver algunas de las imágenes que se utilizan en la etapa de entrenamiento, posterior al proceso de preparación de los datos.



**Fig. 2.** Imágenes procesadas y listas para ser utilizadas en el entrenamiento de la Red Neuronal.

**2.3 Entrenamiento de la Red Neuronal Convolutacional (CNN)**

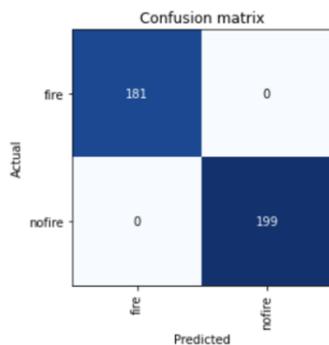
Una vez que se prepararon los datos, se debe entrenar la red neuronal. Se probaron 2 modelos que reciben los nombres de *ResNet* y *ConvNeXt*.

Se crearon y probaron 3 versiones de *ResNet*: *ResNet34*, *ResNet50* y *ResNet101*. En los 3 casos, el modelo arrojó falsos positivos y negativos en imágenes de bosques donde había neblina, mucho humo, o imágenes donde se veía el atardecer de fondo, como se observa en la **Fig. 3**



**Fig. 3.** Ejemplo de falsos positivos y negativos obtenidos usando *ResNet*

Para mejorar los resultados obtenidos con *ResNet*, se creó una red CNN denominada *ConvNeXt* que obtuvo una precisión del 100%, que significa que no hubo falsos positivos o negativos, y el modelo pudo clasificar correctamente las imágenes con y sin fuego. En la **Fig. 4** se observa la matriz de confusión asociada al modelo *ConvNeXt*



**Fig. 4.** matriz de confusión del modelo *ConvNeXt*

Después de generar los 4 modelos y comparar los resultados de la matriz de confusión de cada uno, se decidió integrar el modelo *ConvNeXt* en el tablero de la aplicación de monitoreo. Es decir, se priorizó un modelo que no genera falsos negativos, porque se desea que el modelo no falle cuando se necesite identificar si en una imagen se está produciendo un incendio, que se traduce en el envío de alertas de incendios en fases tempranas del incidente.

Con la información provista por el modelo, en el tablero de monitoreo se genera una alerta que le permite a los equipos de bomberos reaccionar a tiempo, y así reducir las probabilidades de que el fuego se convierta en un desastre difícil de contener.

---

### 3 Posibles Líneas de Investigación Futura

Como línea de investigación futura, proyectada a desarrollar dentro del INTEC, se propone extender al presente monitoreo por imágenes, con una red de dispositivos IoT con sensores de variables meteorológicas y atmosféricas. Este tipo de monitoreo por sensores, que incluiría variables climáticas, como la humedad relativa y la temperatura, y del estado del aire, como la presencia de partículas de humo y de gases combustibles, [9] permitiría obtener indicios de la presencia de incendios en la zona.

Para validar el aporte del presente trabajo y sus líneas de investigación, se entrevista a una autoridad de la Federación Mendocina de Bomberos Voluntarios. Respecto a StopFire, el entrevistado indica que mejoraría el esquema actual de prevención de incendios, ya que dependen principalmente de notificaciones vía telefónica y del análisis de imágenes satelitales de sitios gratuitos y en tiempo real como *FireMap* [10].

En cuanto al aporte de la solución de monitoreo integral, el entrevistado la considera *muy buena*, y agrega que es ideal detectar la columna de humo ‘cuanto antes’ porque las dos primeras horas son cruciales para controlar un incendio y evitar el caos.

**Agradecimientos.** Al Instituto de Tecnología (INTEC) de la Universidad Argentina de la Empresa (UADE) por integrar el PFI en la línea de investigación de *Aplicaciones de Machine Learning para Mejorar el uso de Recursos Naturales* (A21T03).

### Referencias

1. Argentina.gov.ar, «¿Qué es y cómo funciona el Servicio Nacional de Manejo del Fuego?», (2020). <https://www.argentina.gov.ar/ambiente/fuego/servicio-nacional> (accedido 31/7/22).
2. Argentina.gov.ar, «Reportes diarios del Servicio Nacional de Manejo del Fuego», (2021). <https://www.argentina.gov.ar/ambiente/fuego/diciembre-2021> (accedido 31/7/22).
3. Kurtz, D.B., Perucca, A.R., Saucedo, G., «Al 21 de febrero de 2022, la superficie quemada fue de 934.238 hectáreas | INTA». <https://inta.gov.ar/noticias/al-21-de-febrero-de-2022-la-superficie-quemada-fue-de-934238-hectareas> (accedido 31/7/22).
4. Inchausti, P.E., Martínez Saucedo, A.C., Amet, L., Blanco, P., Nievas, G., Giusto, L., «Aplicaciones de Machine Learning para el uso Sustentable de Recursos Naturales». <https://wicc2022.tk/workshop/6256d0d67c76870009464c77/post/6260a4135fedd100097c5c73> (accedido 31/7/22).
5. Martínez Saucedo, A.C., «AQUA: Desarrollo de un Modelo de Machine Learning para Prevenir Incendios Forestales en Pinamar», Tesis, Universidad Argentina de la Empresa, (2021). <https://repositorio.uade.edu.ar/xmlui/handle/123456789/14106> (accedido 5/9/22)
6. Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., De, D., «Fundamental Concepts of Convolutional Neural Network», (2020), pp. 519-567. doi: 10.1007/978-3-030-32644-9\_36.
7. fast.ai, «fast.ai - Welcome to fastai». <https://docs.fast.ai/> (accedido 31/7/22).
8. Dincer, B., «Kaggle» (2021) <https://www.kaggle.com/datasets/brsdincer/wildfire-detection-image-data> Wildfire Detection Image Data (accedido 31/7/22).
9. Argentina.gov.ar, «¿Cuáles son las variables y qué factores las afectan?», (2018). <https://www.argentina.gov.ar/ambiente/fuego/conocemas/variables> (accedido 3/9/22).
10. Robert E. Wolfe, «NASA-FIRMS», (2022). <https://firms.modaps.eosdis.nasa.gov/map/> (accedido 22/8/22).

### 11.5. Cronograma

Se expone el cronograma que se ejecutó para las entregas preliminares y para la entrega final del proyecto completado.

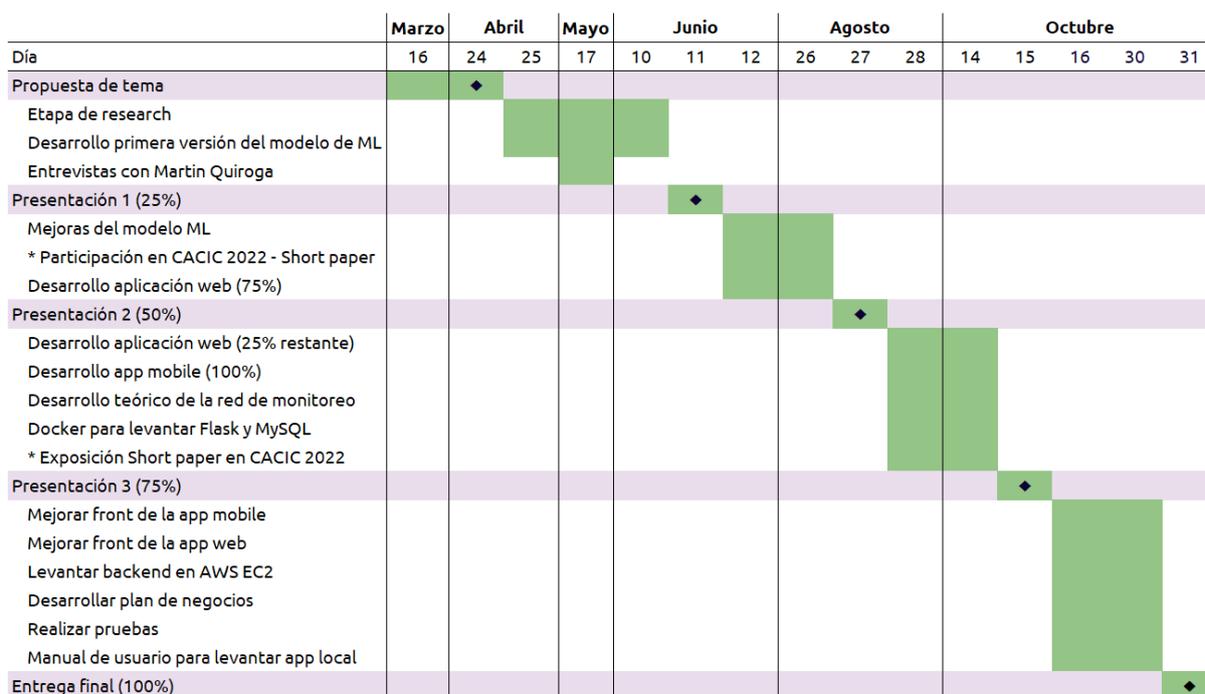


Figura 90: cronograma final

Fuente: elaboración propia