

Patrón para Identificar en la Web de las Cosas los Puntos de Despliegue de Algoritmos de Inteligencia Computacional



Trabajo de grado
Ing. Camilo Enrique Romero Parra

Director: PhD. Miguel Ángel Niño Zambrano
Codirector: PhD. Carlos Alberto Cobos Lozada

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Programa de Maestría en Computación
Grupo de I+D en Tecnologías de la Información (GTI)
Popayán, febrero 2023

Patrón para Identificar en la Web de las Cosas los Puntos de Despliegue de Algoritmos de Inteligencia Computacional

Ing. Camilo Enrique Romero Parra

Trabajo de grado presentado a la Facultad de Ingeniería
Electrónica y Telecomunicaciones de la Universidad
del Cauca para la obtención del Título de:
Magíster en Computación

Director: PhD. Miguel Ángel Niño Zambrano
Codirector: PhD. Carlos Alberto Cobos Lozada

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones Programa de
Maestría en Computación
Grupo de I+D en Tecnologías de la Información (GTI)
Popayán, febrero 2023

Tabla de Contenido

1.	Contextualización	1
1.1.	Introducción.....	1
1.2.	Planteamiento del problema.....	1
1.3.	Objetivos	2
1.3.1.	Objetivo general.....	3
1.3.2.	Objetivos específicos.....	3
1.4.	Resultados obtenidos.....	3
1.5.	Estructura de la monografía	4
2.	Conceptos Teóricos	5
2.1.	Patrones de arquitectura para sistemas IoT.....	8
2.1.1.	Nivel de abstracción	9
2.1.2.	Especificidad de dominio	9
2.1.3.	Requisitos no funcionales.....	9
3.	Revisión sistemática de la literatura	10
3.1.	Introducción.....	10
3.2.	Metodología	11
3.2.1.	Planificación.....	11
3.2.2.	Ejecución	15
3.3.	Resultados	16
3.3.1.	Técnicas de IC utilizadas para el procesamiento de datos.....	16
3.3.2.	Innovación y criterios presentes en la literatura.....	17
3.3.3.	Limitaciones.....	18
3.3.4.	Evaluación de la calidad	20
3.4.	Conclusiones y Trabajos Futuros de la Revisión	21
4.	Propuesta de patrón de arquitectura para la IoT.....	22
4.1.	Aplicación del Patrón de Investigación Iterativa de Pratt.....	22
4.1.1.	Primera iteración para la construcción del patrón.....	22
4.1.2.	Segunda iteración para la construcción del patrón	23
4.1.3.	Tercera iteración para la construcción del patrón	23
4.2.	Patrón arquitectónico propuesto	24
4.2.1.	Nombre.....	24
4.2.2.	Contexto	24
4.2.3.	Problema	25
4.2.4.	Solución.....	26
4.2.5.	Ejemplo de uso del patrón	30
4.2.6.	Implementación en el marco del ecosistema IoT.....	38

5.	Grupo Focal	41
5.1.	Planteamiento de la investigación	41
5.1.1.	Objetivo del grupo focal	41
5.1.2.	Objetivo de la investigación	41
5.1.3.	Preparación de materiales y procedimientos a seguir por parte del grupo de investigación	41
5.1.4.	Protocolo del grupo focal	42
5.1.5.	Elementos necesarios para llevar a cabo el grupo focal.....	42
5.1.6.	Métodos de captura y registro de información	42
5.1.7.	Métodos de análisis de la información	43
5.2.	Diseño del grupo de discusión (Reclutamiento)	43
5.2.1.	Perfil del participante	43
5.2.2.	Identificación de los participantes	43
5.2.3.	Conducción de la sección de debate	43
5.2.4.	Análisis de la información y reporte de resultados.....	45
6.	Prueba de concepto	51
6.1.	Primer caso de estudio.....	51
6.1.1.	Evaluación de viabilidad	51
6.1.2.	Evaluación de las ubicaciones.....	54
6.1.3.	Despliegue.....	58
6.2.	Segundo caso de estudio.....	60
6.2.1.	Evaluación de viabilidad	60
6.2.2.	Evaluación de las ubicaciones.....	63
6.2.3.	Despliegue.....	66
7.	Conclusiones y trabajos futuros	68
8.	Bibliografía	70

Lista de Figuras

Figura 1. Ubicación y procesamiento de datos. Imagen propia.....	16
Figura 2. Gráfico de araña de calificación en el Fog. Imagen propia	36
Figura 3. Gráfico de araña de calificación en el Cloud. Imagen propia	37
Figura 4. Gráfico de araña de comparativa de calificación. Imagen propia	38
Figura 5. Pasos para la implementación del patrón propuesto (Adaptado de CRISP- DM [67]).....	39
Figura 6. Fases de la implementación del patrón propuesto (Adaptado de CRISP- DM [52]).....	40
Figura 7. Gráfico de araña de calificación en el Fog. Imagen propia	56
Figura 8. Gráfico de araña de calificación en el Cloud. Imagen propia	57
Figura 9. Gráfico de araña de comparativa de calificación. Imagen propia	58
Figura 10. Detección de cáncer en cerebro enfermo. Imagen propia.....	59
Figura 11. Detección de cáncer en cerebro sano. Imagen propia	59
Figura 12. Gráfico de araña de calificación en el Fog. Imagen propia	64
Figura 13. Gráfico de araña de calificación en el Cloud. Imagen propia	65
Figura 14. Gráfico de araña de comparativa de calificación. Imagen propia	66
Figura 15. Fotogramas del video generado por el algoritmo. Imagen propia	67

Lista de Tablas

Tabla 1. Número de estudios por base de datos	13
Tabla 2. Limitaciones	18
Tabla 3. Evaluación de la calidad	20
Tabla 4. Evaluación de la importancia por requisito	27
Tabla 5. Evaluación de requisitos no funcionales.....	28
Tabla 6. Descripción de los datos	31
Tabla 7. Evaluación de la viabilidad del despliegue en el Edge	31
Tabla 8. Evaluación de la viabilidad del despliegue en el Fog	32
Tabla 9. Evaluación de la viabilidad del despliegue en el Cloud	33
Tabla 10. Evaluación de la importancia por requisito	35
Tabla 11. Ejemplo de evaluación de los requisitos no funcionales en el Fog.....	35
Tabla 12. Ejemplo de evaluación de los requisitos no funcionales en el Cloud	36
Tabla 13. Protocolo del grupo focal.....	42
Tabla 14. Elementos a tener en cuenta en la realización del Grupo Focal	42
Tabla 15. Perfil profesional de los participantes en el grupo focal.....	43
Tabla 16. Organización del grupo focal.....	44
Tabla 17. Cuestionario de evaluación usado en el grupo focal	44
Tabla 18. Conteo de respuestas a preguntas P1 a P4.....	45
Tabla 19. Respuestas a las preguntas abiertas	46
Tabla 20. Acciones de mejora definidas para el modelo	47
Tabla 21. Cuestionario de evaluación de la entrevista	48
Tabla 22. Conteo de respuestas a preguntas P1 a P4.....	49
Tabla 23. Respuestas a las preguntas abiertas	49
Tabla 24. Acciones de mejora definidas para el modelo	50
Tabla 25. Descripción de los datos	52
Tabla 26. Evaluación de la viabilidad del despliegue en el Fog	52
Tabla 27. Evaluación de la viabilidad del despliegue en el Cloud	53
Tabla 28. Evaluación de la importancia por requisito	55
Tabla 29. Ejemplo de evaluación de los requisitos no funcionales en el Fog.....	55
Tabla 30. Ejemplo de evaluación de los requisitos no funcionales en el Cloud	56
Tabla 31. Descripción de los datos	60
Tabla 32. Evaluación de la viabilidad del despliegue en el Fog	61
Tabla 33. Evaluación de la viabilidad del despliegue en el Cloud	62
Tabla 34. Evaluación de la importancia por requisito	63
Tabla 35. Ejemplo de evaluación de los requisitos no funcionales en el Fog.....	63
Tabla 36. Ejemplo de evaluación de los requisitos no funcionales en el Cloud	64

Lista de Anexos

Anexo 1. Artículo titulado: “Criterios de ubicación de algoritmos de inteligencia computacional en la web de las cosas Revisión sistemática de la literatura”, publicado en publicado en la revista científica *Investigación e Innovación en Ingenierías* de la Universidad Simón Bolívar.

Anexo 2. Resultados detallados del Grupo Focal: Patrón para identificar en la Web de las Cosas los puntos de despliegue de algoritmos de Inteligencia Computacional.

Anexo 3. Resultados detallados Entrevista Focal: Patrón para identificar en la Web de las Cosas los puntos de despliegue de algoritmos de Inteligencia Computacional.

Anexo 4. Iteraciones uno y dos del patrón detalladas.

Resumen

Cuando un desarrollador de soluciones tecnológicas incursiona en el campo de la IoT, encuentra posibilidades de interactuar con diferentes dispositivos y obtener una gran variedad de servicios. Sin embargo, estas capacidades están limitadas tanto por el hardware como por el software de los dispositivos.

Dentro de los ecosistemas IoT, un desarrollador tiene la posibilidad de realizar las tareas de procesamiento no solo en los dispositivos ubicados en el borde de la red, sino también de utilizar los dispositivos concentradores que se encuentran en la niebla y de consumir servicios alojados en el Cloud. Pese a que existe esta posibilidad, se presenta una dificultad a la hora de elegir la ubicación que mejor se adapta a las necesidades específicas del desarrollador y del usuario final del despliegue. Para dar solución a esta necesidad, el presente estudio propone un patrón arquitectónico que permite guiar la elección de los puntos de despliegue de algoritmos de inteligencia artificial que mejor se adapten a las necesidades de la solución tecnológica que se esté desarrollando.

Se presentó una versión preliminar del patrón a un grupo focal compuesto de expertos en el área de IoT, y de metodologías ágiles, los cuales realizaron sugerencias que permitieron obtener la versión final del patrón, el cual fue presentado nuevamente con los expertos recibiendo opiniones favorables frente a los cambios realizados. Finalmente, el patrón propuesto demostró ser de utilidad a la hora de guiar la elección de la ubicación del procesamiento mediante dos pruebas de concepto implementadas en entornos reales.

Dedicatoria

*A mis padres por todo el
apoyo, amor y dedicación
que han tenido conmigo,
gracias por ayudarme a
cumplir mis objetivos como
persona y estudiante.*

Agradecimientos

Gracias a mi familia, en especial a mi padre y a mi madre por todo su cariño, comprensión y ayuda en cada paso de mi vida. Esto es posible gracias a los ánimos que me brindan cada día y a los muchos esfuerzos realizados por darme una buena educación.

Gracias Doctores, Miguel Ángel Niño y Carlos Alberto Cobos, por toda la ayuda y paciencia. Admiro mucho que siempre buscan la excelencia en todo lo que elaboran y por supuesto el amor que le tienen a la academia.

Gracias a todos mis amigos y conocidos por su compañía y cariño.

1. CONTEXTUALIZACIÓN

1.1. Introducción

Dentro del campo de la ingeniería la Internet de las Cosas (Internet of Things – IoT), ha tomado relevancia por las posibilidades que ofrece ante problemas actuales. La IoT permite una interacción entre el mundo físico (compuesto por hardware, sensores y actuadores) y el mundo digital (software con capacidad computacional y de conectividad) embebida dentro de un objeto inteligente [1].

Los dispositivos inteligentes permiten a sus usuarios interactuar y obtener una gran variedad de servicios gracias a su capacidad de procesamiento, la cual se ve limitada por sus capacidades tanto de hardware como de software. Sin embargo, los dispositivos inteligentes no son los únicos que forman parte de un ecosistema inteligente. La computación en la nube (Cloud Computing), en el borde (Edge Computing) y en la niebla (Fog Computing), se unen para permitir que los servicios solicitados puedan tener una mayor velocidad de respuesta, capacidad de procesamiento, seguridad, entre otras.

El presente estudio propone un patrón arquitectónico que permite definir los puntos de despliegue de algoritmos de inteligencia computacional en una solución para un ecosistema inteligente de objetos de la web de las cosas, con el fin de obtener el mejor tiempo de respuesta en el consumo de los servicios soportados.

1.2. Planteamiento del problema

En años recientes se ha producido un gran interés dentro del área del Internet de las Cosas, la cual permite una interacción entre el mundo físico (compuesto por hardware, sensores y actuadores) y el mundo digital (software con capacidad computacional y de conectividad) embebida dentro de un objeto inteligente [1]. Adicionalmente, la Web de las cosas (Web of Things - WoT) permite la integración de objetos inteligentes con servicios que ofrece la Web [2], comunicándose por medio de protocolos bien definidos y permitiendo que un usuario pueda acceder a una gran variedad de servicios y recursos determinados por las necesidades específicas de cada usuario [3][4].

Actualmente, se está llevando el desarrollo de la IoT hacia los ecosistemas de objetos inteligentes de la IoT. Estos ecosistemas consisten en un conjunto de objetos con capacidades de razonamiento, procesamiento y comunicación, que se interconectan entre sí para generar servicios interoperables entre ellos. Uno de los principales desafíos para desarrollar este tipo de sistemas es el problema de la heterogeneidad profunda [5].

Dentro de las ubicaciones de la IoT destaca la computación en el Cloud [6], la cual ofrece la posibilidad de acceder a una amplia variedad de servicios a través de Internet, con la ventaja de que los datos de los usuarios sean accedidos desde cualquier lugar y estar disponibles sin requerir de una instalación de software previa en un dispositivo, reduciendo los costos que esto conlleva.

Una de las principales falencias encontradas en el estado del arte es la falta de distinción entre el Edge y el Fog, siendo términos que los propios autores deciden intercambiar y asumir como iguales [7] [8] [9]. El Edge Computing puede procesar los datos adquiridos en el punto de origen mediante una aplicación IoT, teniendo las capacidades de procesamiento y comunicación directamente dentro del dispositivo inteligente [10], mientras que en el Fog Computing hay un único dispositivo centralizado responsable del procesamiento, que se encarga de analizar los datos de diferentes puntos finales en la red, tomando la información de los dispositivos inteligentes ubicados en el Edge [11].

La ventaja que presenta la computación en el Edge frente a la computación en el Cloud está en los tiempos de respuesta, los cuales pueden reducirse gracias a que el procesamiento se hace de manera local tan pronto como se adquieren los datos por parte del dispositivo inteligente. Por otro lado, la ventaja que presenta la computación en el Cloud frente a la computación en el Fog está en la capacidad de procesamiento y almacenamiento de datos [12]. Dado lo anterior, los diseñadores de aplicaciones IoT tienen inconvenientes al momento de decidir en qué sitio de la red desplegar los diferentes algoritmos de procesamiento de datos. Normalmente toman la decisión de manera empírica, lo cual en muchos casos genera retrasos de desarrollo, o en el peor de los casos, aplicaciones que no cumplen con los requisitos de eficiencia [30] [31].

Por esta razón, es importante encontrar una herramienta, para que los desarrolladores de aplicaciones IoT puedan contar con un conjunto de criterios estandarizados, que permita orientarlos en la decisión de las ubicaciones óptimas para desplegar los algoritmos de procesamiento de datos dentro de un Ecosistema de Objetos Inteligentes de la Web de las Cosas (Intelligent Objects Ecosystem of Web of Things - IOEoWoT), de forma que se puedan aprovechar los beneficios propuestos por las tres ubicaciones mencionadas.

Ante el problema expuesto se planteó la siguiente pregunta de investigación: ¿Cómo determinar la ubicación (edge, fog, o cloud) de los algoritmos de inteligencia computacional en un IOEoWoT, con el fin de obtener el mejor tiempo de respuesta en el consumo de los servicios soportados?

1.3. Objetivos

A continuación, se presentan los objetivos del proyecto tal como fueron aprobados por el consejo de la Facultad de Ingeniería Electrónica y Telecomunicaciones.

1.3.1. Objetivo general

Proponer un patrón arquitectónico que permita definir los puntos de despliegue de algoritmos de inteligencia computacional en una solución para un ecosistema inteligente de objetos de la web de las cosas, con el fin de obtener el mejor tiempo de respuesta en el consumo de los servicios soportados.

1.3.2. Objetivos específicos

- Establecer una revisión sistemática de la literatura en la temática de localización del procesamiento dentro de ecosistemas inteligentes que utilizan algoritmos de inteligencia computacional, mediante las guías propuestas por Kitchenham [13], buscando identificar los enfoques de construcción de patrones arquitectónicos más recientes y exitosos, para establecer los elementos a tener en cuenta en este contexto.
- Definir un patrón arquitectónico para definir la localización del procesamiento de algoritmos de inteligencia computacional dentro de un IOEoWoT, haciendo uso del patrón de investigación iterativa propuesto por Pratt [14] y una retroalimentación de la propuesta a través de un focus group.
- Realizar una prueba de concepto, haciendo uso de un ambiente real o simulado para la planificación de un IOEoWoT en un caso en el cual se precise ubicar adecuadamente los algoritmos utilizados de tal forma que se obtenga el mejor tiempo de respuesta (entre el fog, el cloud y el edge), para comprobar la utilidad del patrón propuesto en una aplicación.

1.4. Resultados obtenidos

A continuación, se resumen los principales resultados de la presente investigación:

- Monografía de la investigación. Corresponde al presente documento y resume el problema, los objetivos planteados para resolver dicho problema, el producto obtenido (patrón) así como la evaluación realizada sobre el trabajo desarrollado. Finalmente, presenta las conclusiones y unas ideas de trabajo futuro que sería de interés desarrollar en el área de investigación.
- Patrón propuesto presentado en el Capítulo 4 del presente documento.
- Un artículo: resultado de la investigación realizada. Corresponde al resultado de una revisión sistemática de la literatura que ha sido publicado en una revista categoría B del PUBLINDEX de Minciencias con la siguiente citación: C. E. Romero Parra, M. Ángel Niño Zambrano, y C. A. Cobos Lozada, «Criterios de ubicación de algoritmos de inteligencia computacional en la web de las cosas: Revisión sistemática de la literatura», Investigación e Innovación en Ingenierías, vol. 9, n.º 3, pp. 108–123, dic. 2021. Con el ISSN: 2344-8652.

1.5. Estructura de la monografía

A continuación, se describe de manera general el contenido y organización de la presente monografía:

CAPITULO 1: INTRODUCCIÓN: Hace referencia al presente capítulo que introduce el tema de investigación, presenta la pregunta de investigación que originó el trabajo, los aportes al problema, también los objetivos (general y específicos) definidos para el proyecto, un breve resumen de los resultados obtenidos y finalmente la organización de la monografía.

CAPITULO 2: CONCEPTOS TEÓRICOS: En este capítulo se presenta un conjunto de conceptos teóricos que forman el contexto teórico de la investigación realizada.

CAPITULO 3: REVISIÓN SISTEMÁTICA DE LA LITERATURA: Este capítulo presenta el estado del arte de trabajos relacionadas con criterios de ubicación de algoritmos de inteligencia computacional en la web de las cosas a través de una revisión sistemática de la literatura. De esta revisión se muestra la metodología utilizada y el resultado obtenido, junto con las conclusiones de esta.

CAPITULO 4: PROPUESTA DE PATRÓN DE ARQUITECTURA PARA LA IOT: En este capítulo se presenta la aplicación del patrón de investigación iterativa de Pratt, la cual da como resultado tres iteraciones del patrón, siendo la tercera iteración la propuesta final para el patrón.

CAPITULO 5: GRUPO FOCAL: En este capítulo se presenta el grupo focal utilizado para conocer la opinión y percepción de expertos en IoT, metodologías ágiles de desarrollo, y programación de entornos en IoT, en cuanto a la pertinencia, idoneidad y completitud del patrón arquitectónico de ubicación del procesamiento en ecosistemas IoT. Se presenta la metodología utilizada para llevar a cabo el grupo focal y el resultado obtenido junto con los ajustes realizados a partir de la retroalimentación realizada.

CAPITULO 6: PRUEBA DE CONCEPTO: En este capítulo se evalúa el patrón propuesto a través de una prueba de concepto implementada en cada posible ubicación para el procesamiento dentro de un ecosistema de objetos inteligentes, comprobando la utilidad del patrón propuesto.

CAPITULO 7: CONCLUSIONES Y TRABAJO FUTURO: En este capítulo se presentan las conclusiones obtenidas al finalizar el trabajo de grado junto con posibles trabajos futuros en la temática de interés de esta investigación.

CAPITULO 8: BIBLIOGRAFIA: Este último capítulo contiene las referencias bibliográficas de los artículos y libros consultados para la realización del proyecto

2. CONCEPTOS TEÓRICOS

A continuación, se presentan los conceptos teóricos necesarios para el desarrollo del presente trabajo investigativo.

Patrones Arquitectónicos: La descripción de los patrones arquitectónicos se basa en el triplete contexto-problema-solución y puede elaborarse más con detalles más ricos, centrándose especialmente en la razón fundamental detrás de una solución. Además, los patrones están destinados a funcionar de forma sinérgica en el contexto de un lenguaje de patrones y tienen numerosas interdependencias entre sí. Finalmente, hay una serie de postulaciones para que una solución califique como patrón. El patrón debe capturar la práctica común (por ejemplo, tener al menos tres usos conocidos) y, al mismo tiempo, la solución del patrón no debe ser obvia. Los patrones deben proporcionar soluciones estéticas, y en la literatura de patrones se acentúa el aspecto humano del software [15].

Patrón de investigación iterativa: La primera definición de patrones de diseño orientados al diseño de software aparece en “Design Patterns: Abstraction and Reuse of Object-Oriented Design” en 1993, en su trabajo Gamma et. al. mencionan como los patrones de diseño identifican, nombran y abstraen temas comunes en el diseño orientado a objetos. Capturan la intención tras el diseño a través de la identificación de objetos, su colaboración, y la distribución de responsabilidades [16]. Utilizando la definición de patrones de diseño dada por Gamma et. al, Pratt propone el patrón de investigación iterativa en su artículo titulado, “Design Patterns for Research Methods: Iterative Field Research” [14]. El patrón de investigación iterativa propuesto consta de los siguientes pasos principales:

- Observar la aplicación de destino
- Identificar los problemas de investigación
- Desarrollar una solución al problema seleccionado
- Probar la solución en el dominio original donde se identificó el problema

A medida que se realizan estas pruebas, se están observando, comenzando así una nueva iteración, ya sea refinando el problema actual o identificando nuevas preguntas que se pueden abordar. Sin múltiples iteraciones, esto degenera en un modelo de desarrollo de cascada monolítico.

Objeto Inteligente: Los objetos inteligentes son entidades del mundo real que están conectados a Internet y pueden ser accedidos vía Web, con la capacidad de recopilar datos crudos a través de sensores y a su vez intervenir en su entorno a través de actuadores [5].

Internet de las Cosas (Internet of Things, IoT): La IoT permite una interacción entre el mundo físico (compuesto por hardware, sensores y actuadores) y el mundo digital (software con capacidad computacional y de conectividad) embebida dentro de un objeto inteligente [1]. La IoT “permite a las personas y cosas, estar conectadas en cualquier momento, en cualquier lugar, con cualquier cosa y con cualquier persona, idealmente usando cualquier camino/red y cualquier servicio” [17].

Web de las Cosas (Web of Things, WoT): La WoT permite la integración de objetos inteligentes con la Web [2], comunicándose por medio de protocolos bien definidos permitiendo que un usuario pueda acceder a una gran variedad de servicios y recursos dependiendo de las necesidades específicas de cada usuario [3][4].

Computación en la nube: El Instituto Nacional de Estándares y Tecnología (National Institute of Standards and Technology - NIST) define la computación en la nube (Cloud Computing) como un modelo para permitir el acceso de red a pedido, conveniente y ubicuo a un grupo compartido de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que se pueden aprovisionar y liberar rápidamente con un esfuerzo mínimo de administración o interacción del proveedor de servicios [18].

Computación en el borde: La computación en el borde (Edge Computing), puede procesar los datos adquiridos en el punto de origen mediante una aplicación [19]. En la computación en el Edge las capacidades de procesamiento y comunicación están directamente dentro del dispositivo inteligente. La computación en el Edge se compone de dispositivos inteligentes, Gateways IoT y servicios alojados en el Cloud [11].

Computación en la niebla: A diferencia de la computación en el Edge, en la niebla (Fog Computing) hay un único dispositivo centralizado responsable del procesamiento, que se encarga de analizar los datos de diferentes puntos finales en la red. La computación en el Edge consta de tres segmentos principales: Dispositivos de IoT, Red de niebla (que recibe datos en tiempo real de dispositivos de IoT utilizando una variedad diversa de protocolos de comunicación y realiza análisis en tiempo real) y un entorno del Cloud (que recibe datos para el almacenamiento de los nodos Fog y realiza análisis para inteligencia empresarial) [11].

Ecosistema de la IoT: Un ecosistema de la IoT combina varias de las capas que componen una arquitectura de la IoT, desde la capa de usuario hasta la capa de conectividad. En la práctica, no es fácil definir una arquitectura genérica, sin embargo, se considera que un ecosistema de IoT se compone, de manera general por las capas de censado y componentes embebidos, capa de conectividad, capa de análisis, capa de gestión de datos, y los componentes finales dentro de la arquitectura. En un ecosistema de IoT típico, los componentes de la arquitectura como son, los dispositivos inteligentes, los sensores y los componentes de terceros,

están conectados a los motores de cómputo o instancias en el Cloud a través de Internet [20].

Inteligencia artificial: Es la ciencia y la ingeniería de fabricar máquinas inteligentes, especialmente programas informáticos inteligentes. Está relacionado con la tarea similar de usar computadoras para comprender la inteligencia humana, pero la inteligencia artificial no tiene que limitarse a métodos que sean biológicamente observables [21].

Aprendizaje de maquina (Machine Learning, ML): ML es una técnica computacional que compara entradas de datos de un sistema con sus correspondientes salidas para aprender su comportamiento y representarlo en un modelo o una lista de patrones. Dentro del Machine Learning (ML), se encuentran algunas aplicaciones como Learning Associations, Supervised Learning (Classification), Regression, Unsupervised Learning, Reinforcement Learning, entre otras [22].

Sistemas expertos (Expert Systems): Los sistemas expertos se desarrollaron como herramientas de investigación para tratar con éxito problemas complejos en un dominio limitado, como el diagnóstico de enfermedades médicas. Los sistemas expertos se basan en la captura o extracción manual de reglas con base en el conocimiento de un experto en un tema, para dar solución a un determinado problema a través de un algoritmo que soporta la toma de decisiones basado en dichas reglas [23].

Inteligencia Computacional (Computational intelligence, CI): La CI es la teoría, diseño, aplicación y desarrollo de paradigmas computacionales basados en situaciones inspiradas en la naturaleza que no son fácilmente interpretables por el ser humano (subsimbólicos). La CI es un campo en evolución y en la actualidad, sus representantes más importantes son las redes neuronales, sistemas difusos y metaheurísticas para la optimización [24].

Redes neuronales (Neural Network): Una red neuronal consta de muchos elementos de procesamiento unidos para formar una red adecuada con funciones de ponderación ajustables para cada entrada. Estos elementos de procesamiento generalmente se organizan en una secuencia de capas con conexiones completas o aleatorias entre capas. El funcionamiento de una red neuronal artificial implica dos procesos: aprendizaje y recuerdo. El aprendizaje es el proceso de adaptar los pesos de conexión en respuesta a los estímulos presentados en el búfer de entrada. La red "aprende" de acuerdo con una regla de aprendizaje que gobierna cómo se ajustan los pesos de conexión en respuesta a un ejemplo de aprendizaje aplicado en las memorias intermedias de entrada. Recordar es el proceso de aceptar una entrada y producir una respuesta determinada por el aprendizaje de la red [25].

Sistemas difusos (Fuzzy Systems): En teoría clásica de conjuntos, un individuo es miembro o no miembro de un conjunto específico en cuestión. Existe una distinción clara entre los miembros y los no miembros de cualquier conjunto, así

como un límite muy preciso que indica si un individuo pertenece o no al conjunto. En la teoría clásica, no es permisible que un individuo esté parcialmente en un conjunto y no lo esté mismo tiempo. Por lo tanto, muchos problemas de aplicación del mundo real no pueden ser descritos y, como resultado, no pueden ser resueltos por la teoría clásica de conjuntos, incluyendo todos aquellos que involucran elementos con solo una pertenencia parcial a un conjunto. Por el contrario, la teoría de conjuntos difusos acepta pertenencias parciales, por lo que, en cierto sentido generaliza la teoría clásica de conjuntos. Los principales campos de investigación en la teoría difusa son los conjuntos difusos, la lógica difusa, y la medida difusa. El razonamiento difuso o razonamiento aproximado es una aplicación de la lógica difusa al procesamiento del conocimiento. El control difuso es una aplicación del razonamiento difuso al control. Aunque la mayoría de las aplicaciones de la teoría difusa han estado sesgadas hacia la ingeniería, estas aplicaciones han llegado recientemente a otras disciplinas, como el diagnóstico médico, la psicología, la educación, la economía, la gestión, la sociología, entre otras [26] [27].

Metaheurísticas (Metaheuristics): El principal inconveniente de los algoritmos utilizados para optimización se encuentra en su falta de generalidad y, en ocasiones, en la dificultad de su implementación. Las metaheurísticas parten de la idea de reducir un poco nuestros requisitos sobre la calidad de las soluciones que se pueden encontrar, a cambio de una flexibilidad en la formulación y solución de problemas que no se pueden desarrollar con otras técnicas. En términos más generales, las metaheurísticas son algoritmos de aproximación que brindan soluciones buenas o aceptables dentro de un tiempo de cómputo apropiado pero que no dan garantías formales sobre la calidad de las soluciones, por no hablar de la optimalidad global. Entre las metaheurísticas bien conocidas y establecidas, se pueden mencionar el recocido simulado, los algoritmos evolutivos, el método de colonias de hormigas y los enjambres de partículas. Los nombres de estos métodos dejan claro que a menudo se inspiran en la observación de algún proceso natural complejo que intentan aprovechar de forma abstracta con el fin de resolver algún problema difícil. Una ventaja de las metaheurísticas es que son lo suficientemente flexibles para incluir el conocimiento del problema cuando están disponibles y pueden tratar con funciones objetivo complejas que a menudo se encuentran en aplicaciones del mundo real [28].

2.1. Patrones de arquitectura para sistemas IoT

En el artículo “Landscape of Architecture and Design Patterns for IoT Systems” del año 2020, Washizaki et. al. realizan una revisión sistemática de la literatura, en la cual buscan describir de manera general el panorama actual de los patrones de diseño y los patrones de arquitectura de la IoT, para identificar deficiencias y sugerir mejoras a la hora de crear nuevos patrones [29]. Dentro de este mismo artículo, se realiza una clasificación de patrones para el diseño y la arquitectura de sistemas IoT en tres categorías dependiendo del nivel de abstracción, la especificidad de dominio, y el atributo de calidad a abordar.

2.1.1. Nivel de abstracción

Un ejemplo de alto nivel de abstracción es la arquitectura en capas para aplicaciones de IoT, en la cual se aborda una arquitectura general de IoT sin ahondar en detalles concretos de la arquitectura. Los patrones de estilo de arquitectura se ubican dentro de esta categoría.

Por otro lado, los patrones de arquitectura buscan encapsular el contexto, problemas recurrentes, y su correspondiente solución, conteniendo más información mientras abordan todo el software o el diseño del sistema en lugar de partes específicas, clasificándose de esta manera como patrones con nivel medio de abstracción. Esta es la principal diferencia entre los patrones de arquitectura y los patrones de diseño, pues los patrones de diseño se enfocan en partes específicas del diseño del sistema.

Los patrones de diseño se pueden categorizar dentro del nivel de abstracción bajo, ya que buscan dar solución a problemas recurrentes de diseño específicos, enfocándose en módulos específicos o partes limitadas, en vez de buscar una solución general para el sistema.

2.1.2. Especificidad de dominio

Para poder entender la aplicabilidad y la reusabilidad del dominio es necesario definir qué tan específico es el dominio del patrón propuesto. Esta clasificación se adapta tanto para patrones de arquitectura como para patrones de diseño.

Los patrones no enfocados a IoT pueden ser usados para diseñar sistemas IoT, así como otros sistemas de software, siempre que los problemas que se buscan solucionar se adapten al patrón propuesto.

Los patrones IoT generales son aplicables para cualquier sistema IoT. Por ejemplo, el patrón publicador / suscriptor.

Los patrones IoT de dominio específico abordan un dominio de problema específico, como por ejemplo en el campo de cuidado de la salud.

2.1.3. Requisitos no funcionales

Todos los patrones de diseño de sistemas y software abordan uno o más requisitos funcionales y no funcionales. Algunos patrones de IoT se enfocan en un solo requisito, mientras que otros abarcan varios de ellos.

3. REVISIÓN SISTEMÁTICA DE LA LITERATURA

Con el fin de tener un mayor entendimiento y caracterizar el estado actual del conocimiento acerca de los criterios de ubicación del procesamiento de datos, utilizando técnicas de inteligencia computacional en un ecosistema de objetos inteligentes de la web de las cosas, se realizó la revisión sistemática de la literatura, titulada “Criterios de ubicación de algoritmos de inteligencia computacional en la web de las cosas: Revisión sistemática de la literatura”. El artículo científico de la revisión sistemática fue publicado en la revista científica *Investigación e Innovación en Ingenierías* de la Universidad Simón Bolívar en el *Volumen 9 Número 3 (2021): Número especial resultados del evento XVI Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento - JIISIC 2021*, con el ISSN: 2344-8652. El artículo publicado se muestra en el Anexo 1.

3.1. Introducción

En años recientes se ha producido un gran interés dentro del área del Internet de las Cosas, la cual permite una interacción entre el mundo físico (compuesto por hardware, sensores y actuadores) y el mundo digital (software con capacidad computacional y de conectividad) embebida dentro de un objeto inteligente [1]. Adicionalmente, la Web de las cosas permite la integración de objetos inteligentes con servicios que ofrece la Web [2], comunicándose por medio de protocolos bien definidos y permitiendo que un usuario pueda acceder a una gran variedad de servicios y recursos. Esto depende del contexto determinado por las necesidades específicas de cada usuario [3][4].

Actualmente, se está llevando el desarrollo de la IoT hacia los ecosistemas de objetos inteligentes de la IoT [20]. Estos ecosistemas consisten en un conjunto de objetos con capacidades de razonamiento, procesamiento y comunicación, que se interconectan entre sí para generar servicios interoperables entre ellos. Uno de los principales desafíos para desarrollar este tipo de sistemas es el problema de la heterogeneidad profunda [5].

Dentro de las ubicaciones en la IoT destaca el Cloud Computing [6], la cual ofrece la posibilidad de acceder a una amplia variedad de servicios a través de Internet, con la ventaja de que los datos provistos por dispositivos y usuarios sean accedidos desde cualquier lugar y estar disponibles sin requerir de una instalación de software previa en un dispositivo, reduciendo los costos que esto conlleva.

Dadas las ventajas y desventajas que presentan las distintas ubicaciones de un ecosistema de objetos inteligentes, los diseñadores de aplicaciones IoT tienen inconvenientes al momento de decidir en qué sitio de la red desplegar los diferentes algoritmos de procesamiento de datos. Normalmente toman la decisión de manera

empírica, lo cual en muchos casos genera retrasos de desarrollo, o en el peor de los casos, aplicaciones que no cumplen con los requisitos de eficiencia [30] [31]. Por esta razón, sería importante encontrar una herramienta, para que los desarrolladores de aplicaciones IoT puedan contar con un conjunto de criterios estandarizados, que permita orientarlos en la decisión de las ubicaciones óptimas para desplegar los algoritmos de procesamiento de datos dentro de un Ecosistema de Objetos Inteligentes de la WoT, de forma que se puedan aprovechar los beneficios propuestos por las tres ubicaciones mencionadas.

La presente revisión, tiene como objetivo determinar y caracterizar el estado actual del conocimiento acerca de los criterios de ubicación del procesamiento de datos, utilizando técnicas de Inteligencia Computacional (IC) en un Ecosistema de Objetos Inteligentes de la WoT.

A continuación, en la Sección 3.2 se presenta el protocolo de investigación utilizado para que la revisión sea formal y objetiva. Posteriormente en la Sección 3.3 se realiza el análisis de la información obtenida mediante tablas comparativas, en la cual se identifican limitaciones, innovación, ubicaciones y demás elementos que sean relevantes para tomar la decisión de que algoritmo implementar y en dónde ubicarlo en la arquitectura IoT. Finalmente, en la Sección 3.4 se presentan las conclusiones de la revisión.

3.2. Metodología

Las revisiones sistemáticas, también conocidas como estudios de alcance, permiten obtener una visión general de un área de investigación con el fin de establecer la cantidad de evidencia sobre un tema específico. Mediante las revisiones sistemáticas se pueden determinar áreas donde un estudio primario es más apropiado. La revisión sistemática que se presenta a continuación se basa en los estudios realizados por Petersen [32], el cual a su vez se basó en Kitchenham [13].

3.2.1. Planificación

Dentro de la etapa de planificación se describen todos los pasos realizados para llevar a cabo la búsqueda y el análisis de los estudios tomados en cuenta para esta revisión sistemática. Primero, se identifican las preguntas de investigación que se intentan responder por medio de la revisión sistemática. Segundo, se aplica el método PICOC para identificar las palabras clave que se utilizarán en la cadena de búsqueda. Y finalmente, se especifican los motores de búsqueda utilizados, junto con la cadena de búsqueda y se muestra el número de resultados obtenidos mediante la búsqueda.

3.2.1.1. Preguntas de Investigación

El objetivo de la revisión sistemática es determinar ¿cuáles son los criterios que se deben tener en cuenta a la hora de determinar la ubicación (Edge, Fog, o Cloud) de los algoritmos de inteligencia computacional en un IOEoWoT para realizar el

procesamiento de datos? Con esto en mente, se plantean las siguientes preguntas de investigación (Research Questions RQs):

- RQ1: ¿Qué técnicas de inteligencia computacional son utilizadas para realizar la toma y procesamiento de datos dentro de los ecosistemas inteligentes que se mencionan en cada estudio?
- RQ2: ¿Qué mejora presentan los criterios frente a las técnicas preexistentes en la literatura?
- RQ3: ¿Cuáles son los criterios de localización del procesamiento de datos utilizando técnicas de inteligencia computacional?
- RQ4: ¿Cuáles son las principales limitaciones que presentan los estudios a la hora de localizar el procesamiento de datos dentro de un IOEoWoT?

Una vez realizado el análisis, se busca identificar las carencias existentes en el área para el planteamiento de estudios futuros.

3.2.1.2. Método PICOC

El modelo PICOC (Población Intervención Comparación Salidas y Contexto) sugerido por Kitchenham [13], fue desarrollado para identificar las palabras clave y formular la cadena de búsqueda a partir de las preguntas de investigación.

- Población: En el caso de la localización del procesamiento de datos en un ecosistema inteligente, se refiere a estudios que, mediante el uso de técnicas de inteligencia computacional buscan dar solución a un escenario de aplicación. En el caso de la presente revisión sistemática, se buscan estudios que realicen el análisis de los datos priorizando las capacidades de procesamiento de los recursos hardware y software presentes en el IOEoWoT.
- Intervención: Se refiere a la aplicación de técnicas de inteligencia computacional y el desempeño obtenido al realizar dicho procesamiento en diferentes ubicaciones del ecosistema (diferentes dispositivos hardware y software). En el caso del presente estudio, se busca determinar los criterios que permitan localizar un procesamiento específico.
- Comparación: En este estudio se comparan los criterios aplicados para ubicar el procesamiento dentro del ecosistema y las debilidades que se pueden identificar. Dentro de los criterios que proponga cada estudio, se espera encontrar una comparativa del desempeño alcanzado por la técnica utilizada en las diferentes ubicaciones (Cloud, Fog o Edge).
- Salidas: No se estipulan salidas como objetivo de esta revisión sistemática. El objetivo del presente estudio es realizar una revisión de la literatura existente referente al tópico de interés, y clasificar las soluciones existentes.
- Contexto: Se busca identificar qué criterios se han propuesto para mitigar el problema que se presenta a la hora de querer ubicar el procesamiento de los datos cuando se implementan técnicas de inteligencia computacional soportadas en servicios alojados en el Cloud, en el Fog o en el Edge del ecosistema.

Las palabras clave identificadas son: Web de las cosas (WoT), internet de las cosas (IoT), Fog Computing, Edge Computing, inteligencia computacional, Offloading (y sus sinónimos: localización y ubicación).

3.2.1.3. Estrategia de Búsqueda

Con base en las preguntas de investigación planteadas y las palabras clave identificadas a partir del método PICOC, se estipula la siguiente cadena de búsqueda, la cual fue utilizada en los motores de búsqueda: ScienceDirect, Google Scholar y Springer.

("Fog computing") AND ("Edge computing") AND ("Internet of Things") AND ("Context awareness") AND ("Artificial Intelligence") AND ("Offloading" OR "Location" OR "Placement")

Cabe resaltar que los estudios que incluyen al Fog Computing y Edge Computing derivan, como se dijo anteriormente, del Cloud Computing, razón por la cual no es necesaria su integración en la cadena de búsqueda.

Adicionalmente, esta cadena de búsqueda se adaptó para el motor de búsqueda IEEE de la siguiente manera:

("All Metadata": Cloud computing) AND ("All Metadata": Internet of Things) AND ("All Metadata": Context awareness) AND ("All Metadata": Artificial Intelligence)

Con el fin de llevar un registro ordenado de las referencias utilizadas en la presente revisión, se utilizó el gestor de referencias Mendeley. Éste estudio se ha realizado durante el último trimestre de 2020 y el primer trimestre de 2021. En la Tabla 1 se puede apreciar el número de resultados obtenidos al aplicar la cadena de búsqueda en las bases de datos seleccionadas.

Tabla 1. Número de estudios por base de datos

Bases de datos	Resultados de búsqueda
Google Scholar	406
IEEE	13
ScienceDirect	173
Springer Link	2

3.2.1.4. Criterios de Selección de Estudios Primarios

Los estudios fueron seleccionados a partir de su título y resumen, así como, la lectura de texto completo y la evaluación de calidad (rigurosidad, metodología de investigación implementada, objetividad en el análisis de resultados y limitaciones tenidas en cuenta). La revisión sistemática fue realizada por un único autor, siendo esta una limitación asociada a la confiabilidad de la revisión. Previo a la primera revisión de literatura se identificaron artículos tipo revisión de literatura (Surveys), asociados con las palabras clave identificadas, con el fin de tener claridad frente al tema a tratar y hacia el objetivo propuesto. Frente a dudas que se presentaron conforme se llevaba a cabo la primera revisión de la literatura, se procedió a realizar

una lectura más rigurosa, mejorando la calidad de la exclusión de estudios. Se estipuló una ventana de tiempo de 5 años (desde el año 2016 hasta el 2021), para realizar la búsqueda de estudios, dado que la temática a tratar se encuentra en constante evolución y es un campo muy reciente dentro del contexto de ecosistemas de objetos inteligentes de la WoT.

Los criterios de inclusión que debían cumplir los estudios para ser tomados en cuenta fueron los siguientes:

- Los estudios están dentro del campo de la localización del procesamiento de datos utilizando técnicas de inteligencia computacional en ecosistemas inteligentes con conocimiento del contexto.
- Los estudios proponen criterios claros para realizar la localización del procesamiento de datos.
- Los estudios se basan en ecosistemas inteligentes que consumen servicios alojados en el Cloud y sus derivados.
- Los estudios hacen uso de dispositivos inteligentes para realizar la toma de datos.

Los criterios de exclusión que se tomaron en cuenta son los siguientes:

- El estudio no se encuentra en idioma inglés.
- El estudio no pertenece a la categoría de artículo científico.
- El estudio se encuentra duplicado o desactualizado.
- El estudio no trata el tema de interés.

3.2.1.5. Criterios de Evaluación de Calidad

Con el fin de evaluar la calidad de los estudios incluidos se plantearon las siguientes preguntas, con su correspondiente rango de valoración:

1. ¿Los criterios de localización propuestos fueron puestos a prueba en un caso de estudio?

- Si – 5.0
- No – 0.0

2. ¿Los criterios de localización propuestos se basa en una comparativa clara de un mismo procesamiento realizado en distintas ubicaciones de un mismo ecosistema?

- Efectivamente – Entre 4.0 y 5.0
- Parcialmente - Entre 2.0 y 4.0
- No se realiza comparativa 0.0

3. ¿Cuál fue el número de citas que tuvo el artículo?

Para artículos publicados durante 2021 no se tiene en cuenta el criterio de evaluación, debido a que al ser demasiado recientes tendrán pocas o nulas citas.

Para artículos publicados entre 2018 y 2020:

- De 10 a más citasiones – Entre 4.0 y 5.0
- Entre 5 y 10 citasiones – Entre 2.0 y 4.0
- Entre 0 y 5 citasiones - Entre 0.0 y 2.0

Para artículos publicados entre 2015 y 2018:

- De 20 a más citasiones – Entre 4.0 y 5.0
- Entre 10 y 20 citasiones – Entre 2.0 y 4.0
- Entre 0 y 10 citasiones - Entre 0.0 y 2.0

4. ¿Se realiza una comparación clara de los resultados obtenidos al aplicar métodos propuestos por otros autores?

- De manera clara – Entre 2.5 y 5.0
- De manera ambigua - Entre 0.0 y 2.5

Con el fin de categorizar como los estudios solucionan las preguntas de investigación planteadas, se buscará agrupar la manera en que los estudios den su repuesta a cada pregunta. En caso de no presentarse una generalidad, se realizará el análisis de las respuestas dadas y se destacarán las principales aportaciones realizadas.

3.2.1.6. Métodos de Síntesis

Los resultados obtenidos al categorizar los estudios primarios seleccionados se muestran en la Sección 3.3 mediante la respuesta específica que plantea cada estudio a las preguntas de investigación propuestas.

3.2.2. Ejecución

Cómo se estipuló en la sección anterior, los pasos que se llevaron a cabo durante la revisión consistieron en:

1. Aplicar la cadena de búsqueda definida anteriormente en los motores de búsqueda seleccionados.
2. Aplicar los criterios de inclusión y exclusión estipulados sobre los resultados obtenidos, a partir del resumen, título y conclusiones de los estudios, pero sin limitarse a estas tres secciones del documento cuándo se requería un mayor entendimiento de este. De esta manera se pasó de tener 594 estudios a tener 48. La principal razón por la cual se realizó la exclusión de estos artículos fue la falta de claridad a la hora de proponer criterios para realizar la localización del procesamiento de datos, al priorizar en objetivos diferentes. Por ejemplo, la disminución en el tiempo de respuesta [33], o la seguridad de los datos [34].
3. Tras leer los documentos por completo, y habiendo aplicado nuevamente los criterios de inclusión y exclusión, se identificaron 16 estudios primarios, descartando 32 estudios que no cumplían con los criterios de inclusión y exclusión.

- Finalmente, se realizó la evaluación de la calidad de cada uno de los estudios primarios y su caracterización dependiendo de la respuesta a la que llegaron respecto a las preguntas de investigación planteadas.

3.3. Resultados

3.3.1. Técnicas de IC utilizadas para el procesamiento de datos

En la Figura 1, se muestra de manera gráfica el número de estudios que tienen en cuenta el Cloud, Fog y/o Edge Computing como una posible ubicación para el procesamiento. De igual forma se muestra el número de estudios que utilizan técnicas de inteligencia computacional para realizar el procesamiento de los datos dentro del ecosistema inteligente Procesamiento con Inteligencia Computacional (PIC), el número de estudios que comparan el rendimiento de diferentes técnicas de inteligencia computacional Comparativa de Inteligencia Computacional (CIC), y el número de estudios que cuentan con una implementación de los criterios propuestos mediante una aplicación IoT.

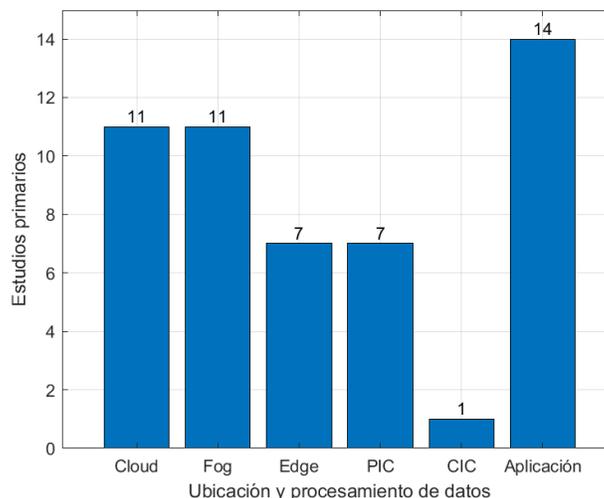


Figura 1. Ubicación y procesamiento de datos. Imagen propia

Como se puede apreciar, un único estudio primario realiza la comparativa de distintas técnicas de Inteligencia Computacional (IC) [8] y siete de ellos utilizan dichas técnicas para realizar el procesamiento de datos en soluciones particulares [35][36][37][8][38][39][40].

Adicional a los 7 artículos que realizaron procesamiento de datos con técnicas de IC, se presentan 3 artículos que, si bien no presentan una aplicación del método propuesto, hacen referencia a la posibilidad de utilizar estas técnicas dependiendo del escenario de aplicación [41][42][30].

Dentro de las técnicas de inteligencia computacional utilizadas predominan el Machine Learning (ML) y el Deep Learning (DL). Además, algunos estudios utilizan metaheurísticas, sin embargo, se evidencia una tendencia al uso de ML y DL.

Dentro de los artículos estudiados se puede apreciar una tendencia al uso de procesamiento distribuido, el cual tiene como objetivo repartir la carga computacional entre los dispositivos que conforman la arquitectura, aprovechando las capacidades de procesamiento de las distintas ubicaciones. Sin embargo, el procesamiento individual que pueda realizar cada dispositivo se deja de lado frecuentemente, siendo esto una limitación, dado que no en todos los casos resulta eficiente repartir la carga computacional.

3.3.2. Innovación y criterios presentes en la literatura

Respondiendo la pregunta de investigación (RQ2), la mayor innovación que presentan los estudios primarios se basa en la optimización del procesamiento dentro de las ubicaciones mediante la descentralización del procesamiento, utilizando el Cloud, el Fog y el Edge Computing, de manera que se aproveche el potencial de procesamiento disponible dentro del ecosistema. De esta manera los estudios logran obtener mejores tiempos de respuesta al solicitar un servicio, mejoras en el consumo energético y mayor seguridad.

Respondiendo la pregunta de investigación (RQ3), dentro de los estudios primarios estudiados se encontró, que los criterios de ubicación del procesamiento se basan en tres (3) aspectos fundamentales: Las características del nodo, las tareas asignadas, y las características de la red. Los dispositivos inteligentes tendrán características hardware relacionadas con la capacidad de procesamiento, la capacidad de almacenamiento y la capacidad de interactuar con sus semejantes estableciendo una comunicación colaborativa. Las tareas variarán en el tiempo requerido para ser desarrolladas y en las características propias de cada tarea, como puede ser su complejidad y los requerimientos del dispositivo para cumplirla.

Se observa que, algunos estudios categorizan las tareas con el fin de asignar un determinado número de tareas a los dispositivos en el Edge, a los nodos Fog, y al Cloud. Otros estudios proponen algoritmos enfocados a la distribución de las tareas de manera automática. Las características de la red se fundamentan en su elasticidad, refiriéndose este término a la escalabilidad de la red, y a su eficiencia, la cual puede variar dependiendo de la infraestructura sobre la cual se soporte el servicio.

Los resultados obtenidos demuestran la necesidad de continuar realizando estudios en el área de la localización del procesamiento en ecosistemas inteligentes. Además, se evidencia una necesidad en hacer un mayor énfasis en la comparativa del rendimiento obtenido en distintas ubicaciones.

3.3.3. Limitaciones

En la Tabla 2. Limitaciones se muestran las principales limitaciones presentes en cada estudio, referentes al posicionamiento del procesamiento dentro de cada ecosistema inteligente, las técnicas de inteligencia computacional (IC) mencionadas dentro del estudio, categorizadas como ML, DL, Federated Learning (FL), Metaheuristics (MH), y Expert System (ES); la comparación del desempeño obtenido por distintas técnicas de inteligencia computacional implementadas en distintas ubicaciones del ecosistema inteligente (CDU), así como limitaciones propias de cada estudio, dando respuesta a la pregunta de investigación (RQ4).

Tabla 2. Limitaciones

Estudio	Ubicaciones estudiadas			Técnicas de IC					CDU	Otras limitaciones
	Cloud	Fog	Edge	ML	DL	FL	MH	ES		
[35]		X		X						No se tiene en cuenta la comunicación con el Cloud
[12]	X	X	X							Sin limitaciones adicionales identificadas
[36]		X		X	X					No se especifica en que consiste la tarea asignada
[43]	X	X								No se especifica en que consiste la tarea asignada No se toman en cuenta las características hardware
[44]	X		X							Sin limitaciones adicionales identificadas
[38]	X	X								Se plantea la posibilidad de usar machine Learning mas no se implementa
[41]			X	X			X			Las tareas que cumplen los dispositivos se limitan a comunicar información y a él envío de ordenes
[42]	X						X			Sin limitaciones adicionales identificadas
[45]	X		X							Los conceptos de Edge y Fog no se diferencian claramente No reduce la carga de transporte de datos
[8]	X	X		X	X				X	Sin limitaciones adicionales identificadas
[37]			X	X	X					Los conceptos de Edge y Fog no se diferencian claramente

[39]		X	X	X							Los conceptos de Edge y Fog no se diferencian claramente No se realiza una clasificación exhaustiva del hardware necesario
[46]	X	X									No tiene claridad en el concepto de Fog, diferenciándolo con servidores MEC
[47]	X	X									Sin limitaciones adicionales identificadas
[40]	X	X		X			X				No tiene claridad en el concepto de Edge
[30]	X	X	X	X							Sin limitaciones adicionales identificadas

Como se puede apreciar en la Tabla 2. Limitaciones, la mayoría de los estudios se enfocan en una computación apoyada en servicios alojados en el Cloud y en el Fog, mientras que menos de la mitad de los estudios se enfocan en llevar el procesamiento hacia el Edge.

Los estudios no tienen un enfoque comparativo entre el desempeño obtenido al realizar el procesamiento de datos entre distintas ubicaciones del ecosistema inteligente, en su lugar buscan la innovación a través de una mejora en los tiempos de respuesta, un menor consumo energético, entre otras.

Estos objetivos se logran al migrar el procesamiento al Cloud y al Edge principalmente, utilizando en ocasiones los nodos Fog como concentradores de la información adquirida por los dispositivos inteligentes que forman parte de la arquitectura IoT y derivando las tareas verticalmente dentro de la arquitectura.

Otra limitación presente en la literatura está asociada con la seguridad, más específicamente en la falta de verificación de la confiabilidad de los dispositivos inteligentes que interactúan dentro del ecosistema aportando información al Edge del despliegue.

Se evidencia una tendencia al uso de Machine Learning y Deep Learning, dejando de lado otras técnicas de Inteligencia Computacional, demostrando un sesgo por parte de la comunidad científica frente a las técnicas que pueden solucionar diferentes necesidades particulares de las aplicaciones estudiadas.

Se puede apreciar una oportunidad de investigación al tener en cuenta diferentes técnicas de IC, además de ML y DL, tales como: aprendizaje federado, sistemas expertos y lógica difusa, que permitan lograr un mejor desempeño en el contexto de la ubicación del procesamiento de datos en un ecosistema inteligente.

Dentro de los estudios principales seleccionados se identificó un único estudio que realiza una comparativa entre el desempeño obtenido al realizar el procesamiento en las tres ubicaciones propuestas [8], sin embargo, el estudio carece de una comparación clara entre distintas técnicas de inteligencia computacional alojadas en cada ubicación, en su lugar utilizan un único algoritmo de inteligencia computacional basado en Machine Learning y se comparan los porcentajes de mejora en los tiempos de respuesta entre una y otra ubicación.

Una de las mayores limitaciones que presentan los estudios es la falta de claridad en la tarea que se va a desempeñar y del algoritmo de inteligencia computacional que se va a implementar para realizar el procesamiento de los datos. Esta limitación nace a raíz de un enfoque de investigación ajeno a la implementación de técnicas de inteligencia computacional en IOEoWoT.

Al realizar la revisión de la literatura gris obtenida en los distintos motores de búsqueda se identifica una dificultad a la hora de diferenciar el Edge Computing y el Fog Computing, resultando en diferentes definiciones que dificultan la correcta identificación de los estudios primarios. Uno de los casos particulares del uso inadecuado del Edge Computing es el término Mobile Edge Computing (MEC), el cual introduce nodos Edge como concentradores de información en vez de utilizar el Fog Computing.

3.3.4. Evaluación de la calidad

En la Tabla 3 se muestran los resultados obtenidos por cada estudio al aplicar la evaluación de la calidad mencionada en la subsección 3.2.1.5. cómo se puede observar la comparativa entre distintas ubicaciones del procesamiento es aplicada únicamente por el artículo [8], la mayoría de los estudios implementaron uno o más casos de estudio sobre el modelo propuesto.

La mayoría de los estudios carecen de un gran número de citas. finalmente se puede concluir que la mayor calificación coincide con el artículo [8], siendo este identificado como uno de los principales referentes a tener en cuenta a la hora de realizar la ubicación del procesamiento de algoritmos de inteligencia computacional en ecosistemas inteligentes.

Tabla 3. Evaluación de la calidad

Estudio	Caso de estudio	Comparación	Citaciones	Calificación
[35]	5	0	0,0	1,67
[12]	5	0	0,0	1,67
[36]	5	0	2,5	2,5
[43]	5	0	2,0	2,33
[44]	5	0	0,0	1,67
[38]	5	0	5,0	3,33

[41]	5	0	0,0	1,67
[42]	5	0	1,5	2,17
[45]	0	0	0,5	0,17
[8]	5	5	1,0	3,67
[37]	5	0	5,0	3,33
[39]	5	0	0,0	1,67
[46]	5	0	0,0	1,67
[47]	0	0	5,0	1,67
[40]	5	0	5,0	3,33
[30]	5	0	1,0	2,0

3.4. Conclusiones y Trabajos Futuros de la Revisión

En el presente estudio, se pudo determinar y caracterizar el estado actual del conocimiento acerca de los criterios de ubicación del procesamiento de datos utilizando técnicas de inteligencia computacional en un IOEoWoT. Se presentó un protocolo de investigación detallado para realizar la revisión del estado del arte de manera objetiva. Aplicando los criterios de inclusión y exclusión estipulados sobre los resultados obtenidos, se pasó de tener 594 estudios a tener 16 estudios primarios, los cuales fueron revisados por completo.

Dentro de los resultados obtenidos se identificó una tendencia entre los estudios de implementar Machine Learning y el Deep Learning como técnicas de IC, siendo algunos de los algoritmos utilizados: Aprendizaje colaborativo, adaptativo y federado, filtrado de datos, clasificación, analítico, vecinos K-más cercanos, entre otros, dejando de lado otras áreas importantes de la IC como los Sistemas expertos, lógica difusa y metaheurísticas. Se puede observar una tendencia entre los estudios principales de buscar disminuir la carga computacional centralizada del procesamiento de datos utilizando tanto el Cloud como el Edge y Fog, sin embargo, se evidencia que pocos estudios implementan criterios para las tres ubicaciones y solamente uno de ellos compara el procesamiento realizado entre ellas [8].

Se puede apreciar una oportunidad de investigación al tener en cuenta diferentes técnicas de IC, además de ML y DL, que permitan lograr un mejor desempeño en el contexto específico de la ubicación del procesamiento de datos en un ecosistema inteligente.

Los resultados obtenidos demuestran la necesidad de continuar realizando estudios en el área de la localización del procesamiento en ecosistemas inteligentes utilizando técnicas de inteligencia computacional para el procesamiento en distintas ubicaciones. Además, se evidencia una necesidad en hacer un mayor énfasis en la comparativa del rendimiento obtenido al realizar implementaciones teniendo en cuenta distintas técnicas de inteligencia computacional.

4. PROPUESTA DE PATRÓN DE ARQUITECTURA PARA LA IOT

4.1. Aplicación del Patrón de Investigación Iterativa de Pratt

El patrón arquitectónico que se propone en este documento se obtuvo utilizando el Patrón de Investigación Iterativa (PII) de Pratt [14] y la propuesta realizada por Washizaki en la cual se categorizan a los patrones arquitectónicos como patrones de nivel de abstracción medio, dado que buscan el encapsulamiento de una mayor cantidad de información que los patrones de diseño, en este sentido el patrón propuesto tiene este mismo nivel de abstracción.

Por otro lado, el patrón propuesto no solo se enfoca en inteligencia computacional, la cual abarca sistemas difusos, redes neuronales y metaheurísticas; en su lugar se incluyen a todas las líneas dentro de los algoritmos de inteligencia artificial, incluyendo los antes mencionadas y añadiendo el Machine Learning, y los sistemas expertos [48].

A continuación, se presentan las tres iteraciones del PII realizadas. En el Anexo 4 se presentan las iteraciones uno y dos de manera detallada.

4.1.1. Primera iteración para la construcción del patrón

En esta primera iteración se buscó experimentalmente comparar el rendimiento obtenido al ejecutar algoritmos de inteligencia artificial (algoritmos de lógica difusa, algoritmos genéticos, entre otras pruebas) en diferentes ubicaciones de ecosistemas inteligentes en la IoT, realizando pruebas entre dispositivos con distintas capacidades de hardware en cada ubicación (por ejemplo, en el Edge una tarjeta ESP8266 como el dispositivo de menor capacidad de cómputo y la tarjeta Raspberry Pi 4 como el dispositivo con mayor capacidad de cómputo).

Se observó que, para guiar la elección de la ubicación sería conveniente presentar al usuario una comparativa con el tiempo de respuesta obtenido al ejecutar un mismo algoritmo en diferentes ubicaciones, así como descartar la posibilidad de realizar el procesamiento de algoritmos concretos debido a las carencias de los dispositivos utilizados.

Se identificó como problema de investigación la gran variedad y cantidad de dispositivos disponibles para realizar los experimentos planteados, y la forma como se deben implementar las soluciones de inteligencia artificial dependiendo de las necesidades del usuario.

Luego de realizar una comparativa de rendimiento entre dispositivos en distintas ubicaciones con distintos algoritmos de inteligencia artificial, se llegó a la conclusión

de que era necesario un cambio en la forma como se estaba desarrollando el proyecto ya que el enfoque experimental no era viable debido a que esto requeriría implementar un número indeterminado y muy grande de pruebas. Por ejemplo, comparar el rendimiento de diferentes dispositivos (con diferentes configuraciones de hardware y software) utilizando un mismo algoritmo implementado en distintos lenguajes de programación. Igualmente, se evidenció la poca literatura con la que se cuenta para implementar algoritmos de inteligencia computacional en las tarjetas basadas en Arduino.

4.1.2. Segunda iteración para la construcción del patrón

En esta segunda iteración se cambió el enfoque experimental y se buscó una solución general para guiar la ubicación del procesamiento dependiendo de las necesidades del usuario.

Se observó que para seleccionar la ubicación se debían identificar los requisitos funcionales y no funcionales de la aplicación que se quisiera realizar (tiempo de respuesta, efectividad, disponibilidad de los datos adquiridos, capacidad de almacenamiento y procesamiento, seguridad, y escalabilidad).

Se identificó como problema de investigación, la necesidad del usuario por delimitar las posibilidades de ubicación para las aplicaciones que deseara desarrollar y como seleccionar la mejor ubicación dentro de las posibles soluciones.

En este sentido, se planteó describir los escenarios de uso del patrón y sus requisitos de uso. Tomando como referente las características de un patrón de diseño definidas por Gamma et. al [16], junto con las características mencionadas por Bloom et. al [49], se proponen las características del patrón a tener en cuenta, y se crea un borrador del patrón en el cual se estipulan las condiciones que debe cumplir el usuario para utilizar el patrón y se guía el desarrollo con base en la metodología CRISP-DM [67].

Al valorar el patrón que se desarrolló hasta ese momento con un experto, se observó qué era necesario mejorar su nivel de detalle, con el fin de proporcionar una guía para el usuario que de una mayor claridad sobre las ubicaciones que son adecuadas para desplegar el procesamiento y que permitiera elegir cual será la más adecuada para cada escenario de uso.

4.1.3. Tercera iteración para la construcción del patrón

En esta iteración se creó la versión final del patrón arquitectónico de ubicación del procesamiento en ecosistemas IoT, se especifica la metodología utilizada para su creación, se describen los componentes del patrón, y se realizó una prueba de concepto para comprobar la utilidad del patrón propuesto en una aplicación.

Por otro lado, con el fin de probar el patrón propuesto, se utilizó un grupo focal compuesto por tres profesionales en ingeniería en áreas afines a la computación, con estudios de maestría o doctorado, con conocimientos en el Internet de las Cosas o metodologías ágiles de desarrollo, como se muestra en el Capítulo 0, junto con una prueba de concepto del patrón propuesto en el Capítulo 6.

4.2. Patrón arquitectónico propuesto

A continuación, se presentan los componentes del patrón arquitectónico de ubicación del procesamiento propuesto.

4.2.1. Nombre

Definición del Punto de Despliegue de Aplicaciones Basadas en Inteligencia Artificial en la WoT.

4.2.2. Contexto

El patrón está dirigido a desarrolladores que implementan una aplicación IoT y busca orientar la elección de la ubicación del procesamiento y análisis de datos utilizando algoritmos de inteligencia artificial en la nube (Cloud), en la niebla (Fog) o en el borde (Edge). Cabe aclarar que las decisiones finales en el desarrollo dependerán de la experiencia del usuario del patrón.

Dentro de un ecosistema de objetos inteligentes de la web de las cosas, existen distintos requisitos no funcionales (seguridad, tiempo de respuesta, conectividad, escalabilidad, entre otros) a los cuales se puede enfocar el diseño de la arquitectura propuesta. Dependiendo de este enfoque, se puede tomar la decisión de realizar el procesamiento y análisis de datos en una ubicación de la arquitectura [50].

Por otra parte, dependiendo de la técnica de inteligencia artificial (machine learning, neural networks, expert systems, fuzzy logic y metaheuristics) que se necesite implementar para dar solución a un determinado problema, en una aplicación específica, es necesario el uso de componentes hardware con capacidades de procesamiento y almacenamiento mínimas para poder desempeñar la tarea requerida.

Dada la complejidad para lograr un equilibrio adecuado entre los requisitos no funcionales y los objetivos de procesamiento de la aplicación particular, este patrón apoya la toma de la decisión de la ubicación más apropiada (Edge, Fog o Cloud), teniendo en cuenta las necesidades específicas de la aplicación IoT que se quiere implementar.

Dependiendo de la ubicación del procesamiento dentro de un ecosistema de objetos inteligentes, se obtendrán ventajas y desventajas desde el punto de vista del usuario. En el Edge la velocidad de respuesta es mayor, se tienen mayores

capacidades de movilidad, autonomía energética, entre otros. En el Fog se tienen capacidades superiores de autenticación de usuario lo cual influye en la seguridad de la aplicación, capacidades de almacenamiento superiores, capacidad para concentrar dispositivos inteligentes en el Edge, entre otras. Finalmente, en el Cloud existe una mayor capacidad de procesamiento, así como de optimización y simulación [50]. En la Tabla 4 se muestra a manera de ejemplo una comparativa entre las tres posibles ubicaciones con algunos requisitos no funcionales.

Tabla 4. Ventajas y desventajas de las ubicaciones

	Cloud	Fog	Edge
Latencia	Desventaja	Neutral	Ventaja
Distancia entre la ubicación y los dispositivos	Lejos del borde (Desventaja)	Cercano al borde (Neutral)	En el borde (Ventaja)
Almacenamiento	Ventaja	Neutral	Desventaja
Potencia de computo	Ventaja	Neutral dependiendo del dispositivo	Desventaja
Capacidad de movilidad	Desventaja	Neutral	Ventaja
Autenticación de usuario	Ventaja	Neutral	Desventaja

Un ejemplo de un escenario de motivación para hacer uso del presente patrón, podría ser una aplicación IoT enfocada en la traducción del lenguaje de señas a texto mediante visión de máquina y algoritmos de inteligencia artificial, más específicamente mediante una red neuronal. En este ejemplo, se presenta una dificultad para el diseñador de la aplicación a la hora de seleccionar la mejor ubicación para el procesamiento de los datos adquiridos por el sensor (en este caso las posiciones de las manos capturadas con la cámara), dado que la captura de datos y el entrenamiento de la red neuronal pueden tomar mucho más tiempo (en caso de que sea del todo posible) en un dispositivo ubicado en el Edge (cómo puede ser una tarjeta Arduino) en comparación con un dispositivo en otra ubicación. Adicionalmente, en el Cloud se necesitará de una conexión a internet para consumir los servicios de procesamiento necesarios con la ventaja de tener una mayor velocidad en el procesamiento y una posible desventaja en la latencia de la conexión.

4.2.3. Problema

¿Dónde ubico una aplicación software basada en inteligencia artificial en una arquitectura que contempla el Edge, Fog y Cloud en un ecosistema de objetos inteligentes de la WoT?

4.2.4. Solución

Antes de definir la ubicación del algoritmo para su uso (Edge, Fog o Cloud), el desarrollador deberá realizar el entrenamiento del algoritmo. En el caso de las soluciones basadas en sistemas expertos, lógica difusa y metaheurísticas, a pesar de que no se tiene una fase con el nombre de entrenamiento, el desarrollador tiene que construir las reglas o algoritmos con base en la experiencia de los expertos del dominio o el problema concreto que se está resolviendo, que es similar a un entrenamiento de un modelo pero que se hace a mano y no de forma automática desde datos. El entrenamiento normalmente se realiza en la ubicación con las mayores capacidades de procesamiento disponibles, donde se cuenta además con las herramientas necesarias para su desarrollo. El desarrollador deberá tener muy en cuenta que el modelo obtenido del entrenamiento, en el caso de este ejemplo una red neuronal, pueda ubicarse en el lugar de despliegue seleccionado.

Una vez se haya realizado el entrenamiento previo del algoritmo, el usuario (desarrollador o arquitecto de la solución) debe evaluar la viabilidad (factibilidad) de cada ubicación (Edge, Fog o Cloud) para la implementación de la solución. Luego, si dos o más ubicaciones se consideran viables se debe evaluar de acuerdo con el contexto de la solución cuál de las ubicaciones es la mejor alternativa.

Las siguientes preguntas definen la viabilidad (factibilidad) de ubicar el procesamiento en una determinada ubicación. Las preguntas deberán ser respondidas con SI o NO. Se inicia evaluando la viabilidad de ubicar la solución en el Edge, en caso de que una de las respuestas sea negativa, esta ubicación será definida como inviable para realizar el procesamiento. Este proceso se repite luego para la ubicación en el Fog y para el Cloud.

- ¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?
 - Haga un listado de los datos y los atributos de estos que requieren para entrenamiento y uso de la aplicación (Este listado es igual para todas las ubicaciones, no lo repita).
 - Defina donde están cada uno de esos datos (Este listado es igual para todas las ubicaciones, no lo repita).
 - Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.
- ¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?
 - Basado en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.

- ¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?
 - Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.
 - Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.

Una vez definidas las ubicaciones viables, se procede a determinar cuál de ellas es la mejor alternativa. Para lograr esto se debe primero ponderar los requisitos no funcionales de la aplicación (tiempo de respuesta, latencia, escalabilidad, facilidad de entrenamiento del modelo de inteligencia artificial y su actualización, facilidad de adquisición de datos, capacidad de procesamiento, interoperabilidad, autonomía y seguridad) de la solución que se va a implementar. La Tabla 5 muestra en su primera columna los requisitos no funcionales a tener en cuenta y en la segunda su importancia correspondiente, la cual deberá puntuarse sin tener en cuenta la ubicación en la que será desplegada la aplicación.

Tabla 5. Evaluación de la importancia por requisito

Requisito no funcional	Importancia (0 menos importante y 5 más importante)
Tiempo de respuesta	
Despliegue y actualización del modelo o del algoritmo	
Adquisición de los datos	
Procesamiento	
Seguridad	
Movilidad	
Consumo energético	
Costo (se puntúa con base en los recursos disponibles)	

Posteriormente, en la Tabla 6, se presenta la evaluación de requisitos no funcionales. En esta tabla se deberá copiar la misma puntuación de la importancia de la Tabla 5 y se dará una calificación a cada una de las preguntas completas que aparecen en la primera columna (según la ubicación que se desee evaluar) con un valor en una escala de 1 a 5 (donde 1 se refiere al peor desempeño y 5 al mejor desempeño) con base en los resultados que se espera provea el algoritmo al ser ejecutado en una determinada ubicación, esta calificación se asigna con base en la

experiencia previa del desarrollador de la aplicación o con base en reportes encontrados en la literatura, lo cual en los dos casos establece una **calificación estimada**.

Una vez se tiene la puntuación de cada ubicación viable, se multiplica la ponderación de cada atributo por su calificación, se suman dichos valores y al final se divide por la suma de las ponderaciones para obtener la calificación final de cada ubicación. La ubicación que obtenga la puntuación más alta se selecciona como la ubicación más apropiada (ver Ecuación 1).

$$Decisión = ArgMax_{u_j \in U} \left(\sum_{r=1}^8 (I_r \times Cal_r) / \sum_{r=1}^8 I_r \right) \quad (1)$$

Donde,

U = Ubicaciones (Edge, Fog o Cloud)

u_j = Corresponde a cada una de las ubicaciones

r = Índice de cada uno de los 8 requisitos no funcionales

I_r = Importancia dada a un requisito no funcional según la Tabla 5

Cal_r = Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5, según la Tabla 6

Tabla 6. Evaluación de requisitos no funcionales

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
¿El tiempo de respuesta de la solución en esta ubicación es el apropiado para el desarrollador?			
¿El proceso de despliegue y actualización del modelo o del algoritmo en esta ubicación es eficiente y eficaz para el desarrollador?			
¿Se puede hacer la adquisición de los datos desde esta ubicación y aquí se puede desarrollar eficientemente?			
¿Con la capacidad de procesamiento disponible, el tiempo de respuesta es apropiado para la llevar a cabo la tarea?			
¿La seguridad que ofrece la ubicación es suficiente para			

cumplir con los requisitos de la aplicación?			
¿La capacidad de movilidad de los dispositivos en la ubicación satisfacen las necesidades del usuario? [51]			
¿El consumo energético de los dispositivos en la ubicación satisfacen las necesidades de la aplicación? [51]			
¿Qué tan viable es utilizar la ubicación según su costo de despliegue?			
TOTAL ACUMULADO	Suma de la calificación de importancias	--	Suma calificación ponderada
CALIFICACIÓN			Suma calificación ponderada / Suma de la calificación de importancias

A continuación, se explican a detalle los requisitos no funcionales:

Tiempo de respuesta: Es el tiempo resultante de la suma entre el tiempo de adquisición de los datos, procesamiento, entrega de respuesta y latencia. Para valorar este requisito en una determinada ubicación el desarrollador deberá preguntarse si con los dispositivos que tiene disponibles, con el alcance que tienen estos dispositivos, puede cumplirse con un tiempo de respuesta apropiado.

Despliegue y actualización del modelo o del algoritmo: Si la aplicación que se va a ubicar se basa en un modelo que además de desplegarse se debe actualizar con una cierta periodicidad, se debe tener claro que esta tarea se puede realizar en la ubicación seleccionada (eficacia), pero además, que el desarrollo de estos dos procesos se pueda realizar con recursos software, hardware, tiempo y costo, que está en concordancia con el presupuesto del proyecto o de la empresa (eficiencia).

Adquisición de los datos: Es una calificación estimada de que tan eficiente es la recolección (adquirir) los datos requeridos para la ejecución de la aplicación basada en IA. Evaluar esta eficiencia puede incluir el tiempo de recolección desde las diferentes fuentes y el consumo energético con los diferentes recursos disponibles en la ubicación.

Procesamiento: La capacidad de procesamiento disponible (procesador, RAM y almacenamiento externo) en la ubicación permitirá obtener el resultado de la ejecución del modelo o el algoritmo en un tiempo que está dentro del rango de lo esperado por parte del usuario.

Seguridad: La ubicación donde se va a alojar la aplicación cuenta con dispositivos que vienen con herramientas de seguridad integradas, estas están apropiadamente configuradas, las contraseñas para el acceso a los dispositivos están inicializadas y son robustas, la definición de la red protege del acceso no autorizado a los dispositivos, los dispositivos tienen su firmware, sistema operativo y otras aplicaciones actualizadas, protegidas y se actualizan constantemente, las funcionalidades o características que no se usan en los dispositivos están deshabilitadas; con lo que se logra que la aplicación este instalada en un entorno donde la seguridad tiene el nivel requerido por la organización que la va a usar.

Movilidad: El usuario cuenta con la suficiente capacidad de desplazamiento en la ubicación seleccionada para el uso de la aplicación y el desarrollo de la tarea para la cual esta fue diseñada.

Consumo energético: La aplicación al ser instalada en la ubicación seleccionada, contara con la capacidad energética suficiente para ser usada todo el tiempo que sea necesario, o por el contrario necesita baterías u otros mecanismos de consumo energético.

Costo: La compra, reposición o alquiler de equipos y software actualizados en la ubicación seleccionada corresponden con el presupuesto establecido para el despliegue de la aplicación.

4.2.5. Ejemplo de uso del patrón

Un ejemplo de la aplicación de la solución del patrón podría ser el procesamiento de imágenes para la selección automática de piezas defectuosas en una línea de producción. En ese ejemplo se tendría una cámara (dispositivo inteligente) que transmite las imágenes capturadas a un computador (dispositivo concentrador) con un sistema operativo actualizado con parches de seguridad, con procesador Intel Core i5 de 8va generación, sin tarjeta gráfica, con 4GB de RAM, en una fábrica, la cual cuenta con servicios de procesamiento en el Cloud para su intranet y extranet. Los desarrolladores consideran que la aplicación se debe desarrollar con una red neuronal profunda.

Siguiendo las indicaciones de la solución, el desarrollador deberá realizar un entrenamiento con el fin de obtener el modelo de la red neuronal a implementar. Una vez se cuenta con un modelo, que en este caso será compatible con las tres posibles ubicaciones, se pasa a evaluar la viabilidad de ubicar el procesamiento en esas tres ubicaciones.

En la Tabla 7 se listan los datos requeridos para la aplicación, así como el lugar donde se alojan estos datos. Esta información se mantiene para las tres posibles ubicaciones.

Tabla 7. Descripción de los datos

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Haga un listado de los datos y los atributos de estos que requieren para entrenamiento y uso de la aplicación.	<ul style="list-style-type: none"> • Datos requeridos: Imágenes de los productos de la línea de producción. • Datos requeridos para entrenamiento: Imágenes e identificadores de los productos con y sin defectos con las que se va a entrenar el modelo. • Atributos requeridos: Nombre de la imagen con su estado (defectuoso o correcto) y dimensiones de la imagen.
Defina donde están cada uno de esos datos.	<ul style="list-style-type: none"> • Los datos serán adquiridos mediante una fotografía, a la cual se le deberá dar el formato adecuado. • Los datos de entrenamiento se obtienen de una base de datos de la empresa.

A continuación, en la Tabla 8, se realiza la evaluación de viabilidad en el Edge.

Tabla 8. Evaluación de la viabilidad del despliegue en el Edge

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> • Si, los datos están disponibles en la ubicación.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<ul style="list-style-type: none"> • Entorno de desarrollo integrado (IDE) para computador como VSCode o Pycharm (Utilizado para el entrenamiento del modelo). • Entorno de desarrollo integrado para Raspberry Geany (Utilizado para el entrenamiento del modelo).

	<p>Normalmente esta opción no se recomienda.</p> <ul style="list-style-type: none"> • Librerías: OpenCV, Scikit-image, PIL/pillow, NumPy, Mahotas
¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> • Dispositivo inteligente - cámara
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> • El dispositivo no cuenta con las capacidades hardware para realizar el procesamiento ni el almacenamiento de datos. • No existen frameworks o librerías para el desarrollo en el sistema operativo propietario del fabricante del dispositivo.

Como se puede apreciar, esta ubicación no es viable y por lo tanto se descarta.

A continuación, en la Tabla 9, se realiza la evaluación de viabilidad en el Fog.

Tabla 9. Evaluación de la viabilidad del despliegue en el Fog

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> • Los datos no se encuentran en la ubicación, pero son fácilmente adquiribles al conectar la cámara al computador.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución,	<ul style="list-style-type: none"> • Entorno de desarrollo integrado (IDE) para computador como VSCode o Pycharm (Para

y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<p>entrenamiento y ejecución del modelo).</p> <ul style="list-style-type: none"> • Librerías: OpenCV, Scikit-image, PIL/pillow, NumPy, Mahotas
¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> • Computador de escritorio, con un sistema operativo actualizado con parches de seguridad, con procesador Intel Core i5 de 8va generación, sin tarjeta gráfica, con 4GB de RAM.
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> • El dispositivo cuenta con la capacidad hardware necesarias para llevar a cabo el procesamiento y almacenamiento de los datos, así como la capacidad para conectarse a distintos dispositivos de la red. • Existen frameworks disponibles para desarrollar el algoritmo de procesamiento requerido.

Esta ubicación es viable, y por lo tanto pasa a la fase de calificación de requisitos no funcionales.

A continuación, en la Tabla 10, se realiza la evaluación de viabilidad en el Cloud.

Tabla 10. Evaluación de la viabilidad del despliegue en el Cloud

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> • Si, los datos pueden ser enviados al Cloud pues se cuenta con conectividad a internet, por lo que se puede transferir la información desde el computador en el Fog.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones	

requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<ul style="list-style-type: none"> Entorno de programación Google Colab, Pycharm, VSCode, etc. (Para entrenamiento y ejecución del modelo). Librerías: OpenCV, Scikit-image, PIL/pillow, NumPy, Mahotas
¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> Depende de los servidores contratados en la nube específica (AWS, GCP, Azure, etc.), o hosting.
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> La ubicación cuenta con las capacidades necesarias para realizar el procesamiento. Existen herramientas para llevar a cabo el procesamiento.

Esta ubicación es viable, y por lo tanto pasa a la fase de calificación de requisitos no funcionales.

De esta manera se sabe que las ubicaciones viables son el Fog y el Cloud. Para lograr determinar cuál de estas ubicaciones será la mejor alternativa para ubicar el procesamiento se utilizan las Tabla 5 y Tabla 6. La evaluación de la importancia se muestra a continuación en la Tabla 11:

Tabla 11. Evaluación de la importancia por requisito

Requisito	Importancia
Tiempo de respuesta	4
Despliegue y actualización del modelo o del algoritmo	5
Adquisición de los datos	5
Procesamiento	4
Seguridad	1
Movilidad	1
Consumo energético	1
Costo	5

El resultado de la evaluación en el Fog del ejemplo se muestra a continuación en la Tabla 12:

Tabla 12. Ejemplo de evaluación de los requisitos no funcionales en el Fog

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
Tiempo de respuesta	4	5	20
Despliegue y actualización del modelo o del algoritmo	5	5	25
Adquisición de los datos	5	5	25
Procesamiento	4	3	12
Seguridad	1	5	5
Movilidad	1	1	1
Consumo energético	1	1	1
Costo	5	5	25
Total Acumulado	26		114
Calificación			4,384615385

Con el fin de tener una representación gráfica del ajuste que presenta la ubicación evaluada respecto a la importancia de cada requisito no funcional, en la Figura 2 se muestra el grafico de araña de la calificación realizada para el Fog en la Tabla 12.

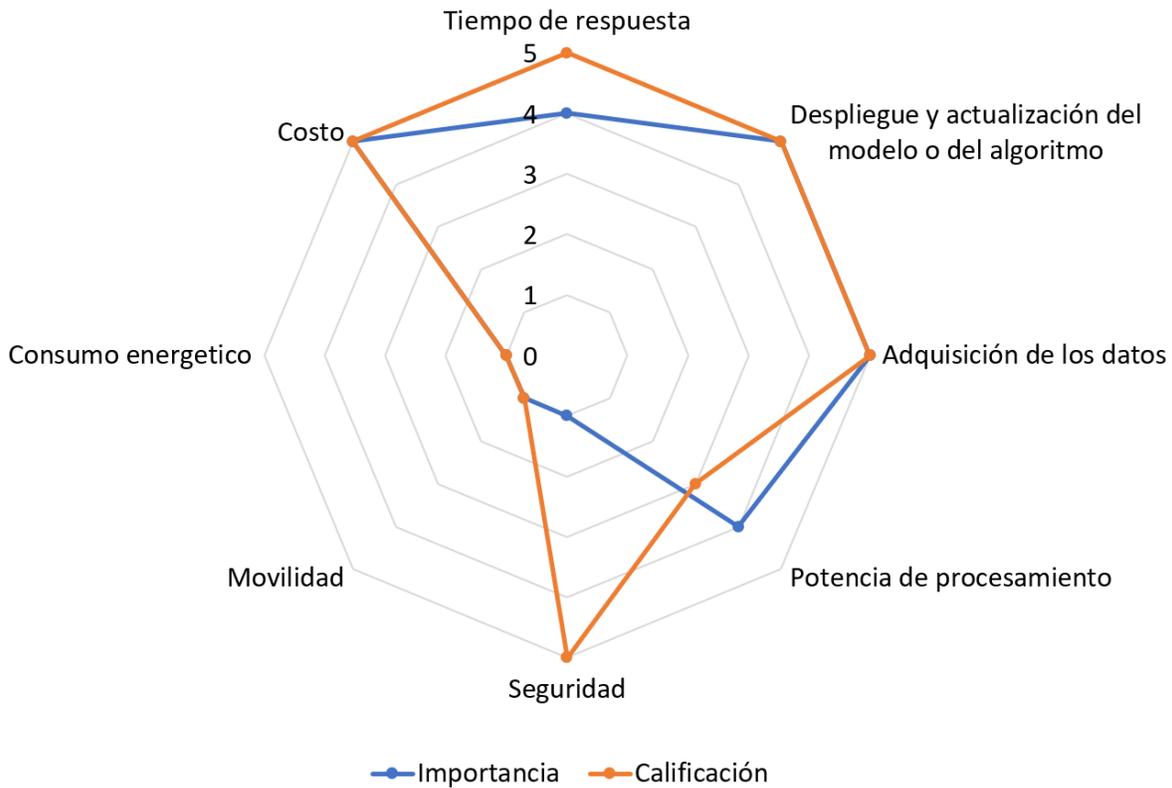


Figura 2. Gráfico de araña de calificación en el Fog. Imagen propia

Como se puede apreciar, la calificación no se ajusta con la importancia del procesamiento, sin embargo, los demás requisitos no funcionales se ajustan adecuadamente.

El resultado de la evaluación en el Cloud del ejemplo se muestra a continuación en la Tabla 13:

Tabla 13. Ejemplo de evaluación de los requisitos no funcionales en el Cloud

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
Tiempo de respuesta	4	4	16
Despliegue y actualización del modelo o del algoritmo	5	3	15
Adquisición de los datos	5	3	15
Procesamiento	4	5	20
Seguridad	1	5	5
Movilidad	1	1	1
Consumo energético	1	1	1
Costo	5	5	25
Total Acumulado	26		98
Calificación			3,769230769

La evaluación realizada en la Tabla 13 se muestra a continuación en la Figura 3:

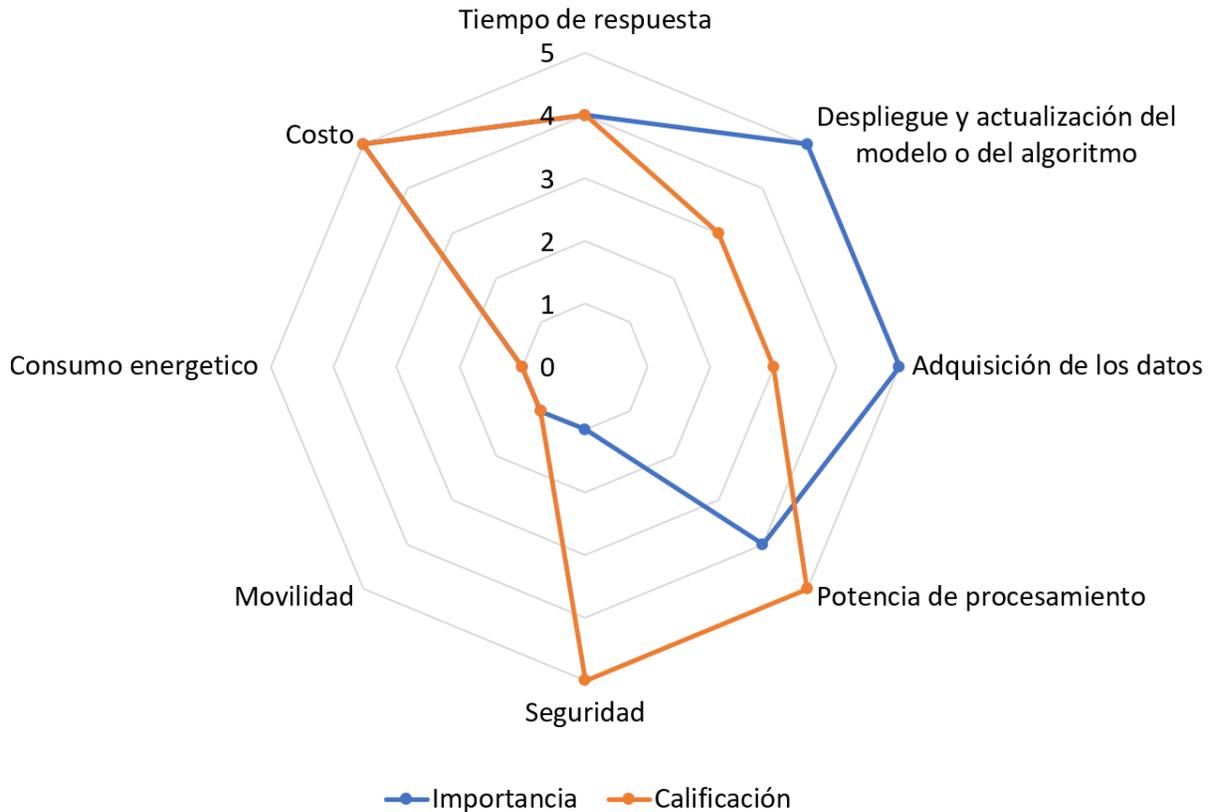


Figura 3. Gráfico de araña de calificación en el Cloud. Imagen propia

Como se puede apreciar, el despliegue y actualización del modelo o del algoritmo y la adquisición de los datos no se ajustan con la valoración de la importancia que se requiere para estos requisitos no funcionales.

A continuación, se presenta en la Figura 4 la comparativa entre las ubicaciones evaluadas.

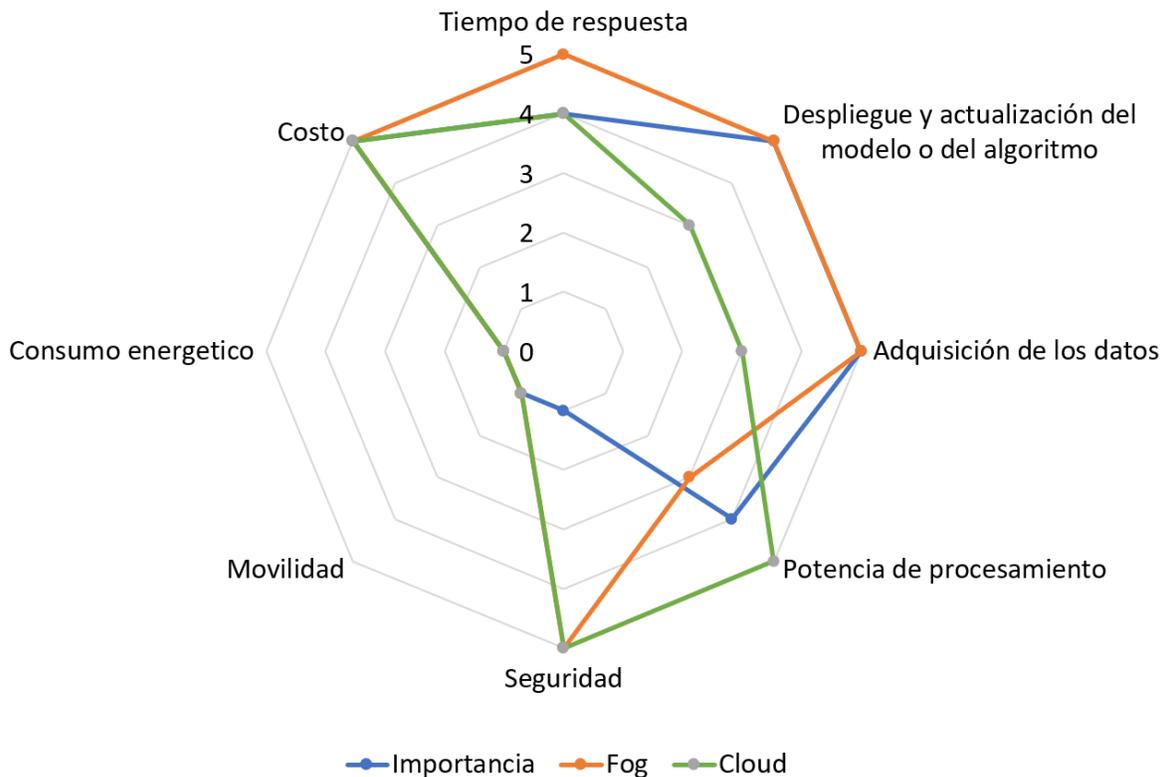


Figura 4. Gráfico de araña de comparativa de calificación. Imagen propia

Finalmente, obtenemos una calificación de 4,38 puntos para el Fog, superando la calificación de 3,77 puntos del Cloud, lo que indica que para la aplicación que se desea implementar el Fog será la mejor alternativa.

4.2.6. Implementación en el marco del ecosistema IoT

Tomando como referencia a CRISP-DM (Cross-Industry Standard Process for Data Mining) [52], el proceso más usado y probado en diferentes contextos de solución de problemas basado en IA y minería de datos en el mundo, se propone la incorporación de una nueva tarea de segundo nivel a la fase de entendimiento del negocio que permite la definición de la ubicación de la aplicación basada en IA usando el patrón arquitectónico propuesto (señalada en la Figura 6).

En la Figura 5 se presentan las fases de la implementación propuesta usando CRISP-DM [67]. La implementación es iterativa, incremental y se estructura en seis fases: (i) entendimiento del negocio, (ii) entendimiento de datos, (iii) preparación de datos, (iv) modelado, (v) evaluación, y (vi) despliegue. Cada fase se compone de tareas que generan uno o más resultados que se usan como entrada de las siguientes fases del proceso.

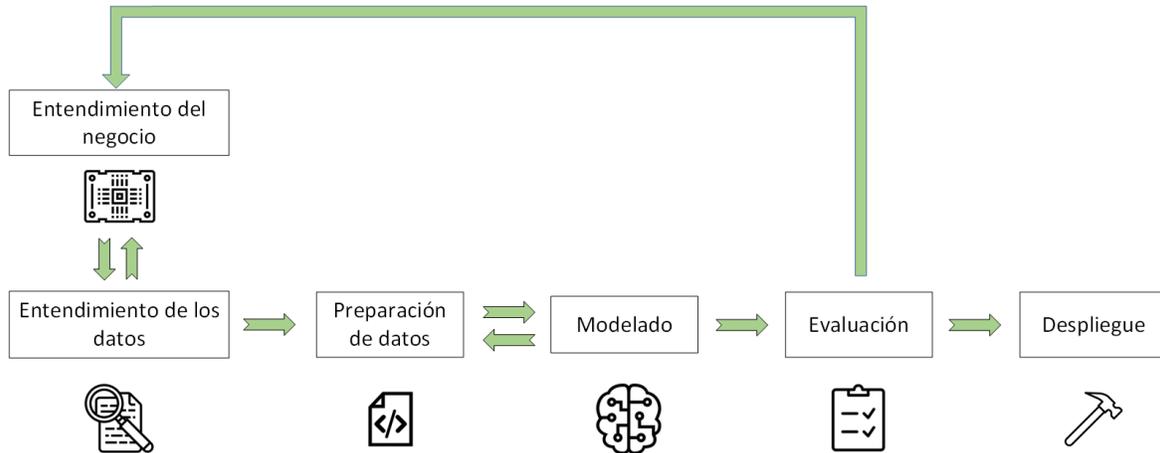
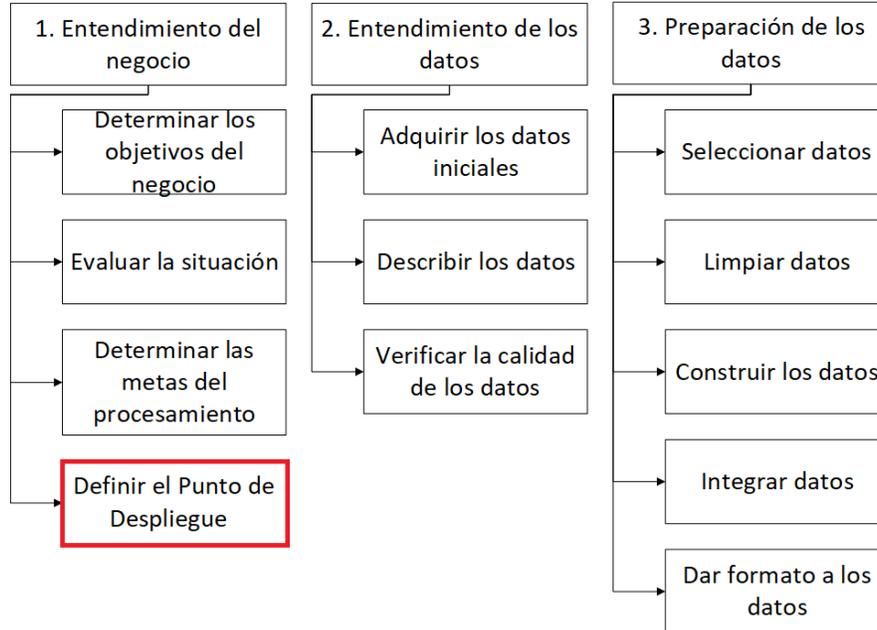


Figura 5. Pasos para la implementación del patrón propuesto (Adaptado de CRISP-DM [67])

El desarrollo de estas fases implica una constante interacción con las fases previas, dado que, el despliegue de una determinada fase puede implicar la creación de una nueva tarea en una fase previa. Es importante entender que el flujo de esta implementación guarda un sentido lógico en la progresión del tratamiento de datos, y regresar a fases que no han sido contempladas dentro del flujo de la estructura podría implicar un retraso en el desarrollo de la aplicación. En la Figura 6 se pueden apreciar las tareas que componen a cada fase.



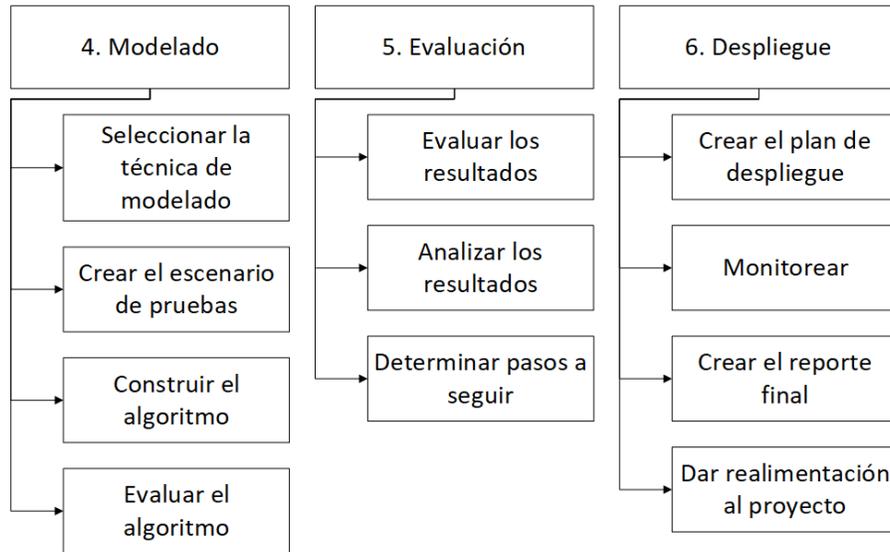


Figura 6. Fases de la implementación del patrón propuesto (Adaptado de CRISP-DM [52])

Como se puede apreciar en la Fase 1 de la Figura 6, se presenta resaltada en rojo la Tarea de segundo nivel propuesta, Definir el Punto de Despliegue; en esta tarea, usando el patrón propuesto y presentado en la Sección 4.2.4, se define la ubicación más apropiada para el despliegue futuro de la solución, con esto además, se logran definir ciertas restricciones que se deben tener en cuenta en el proceso de entrenamiento, modelado o desarrollo de la solución (Fase de modelado, y Fase de evaluación).

5. GRUPO FOCAL

Para realizar el grupo focal se tomaron como referencia los trabajos realizados en [53] y [54]. Los pasos para su elaboración se muestran a continuación:

- **Planteamiento de la investigación:** En esta sección se establecen los elementos que componen al grupo focal.
- **Diseño del grupo de discusión (Reclutamiento):** En esta sección se describen los criterios de selección de los participantes del grupo focal.
- **Conducción de la sesión de debate (Moderación):** En esta sección se ejecutan los procedimientos descritos en la sección de planteamiento, generando una sesión de debate, con el fin de capturar la opinión de los participantes.
- **Análisis de la información y reporte de resultados:** En esta sección se realiza un análisis cualitativo y/o cuantitativo de los resultados obtenidos utilizando estadística descriptiva o métodos de tipo cuantitativo.

A continuación, se llevan a cabo cada una de las secciones mencionadas.

5.1. Planteamiento de la investigación

A continuación, se plantean los elementos que componen al grupo focal:

5.1.1. Objetivo del grupo focal

El objetivo del grupo focal es: conocer la opinión y percepción de expertos en IoT, metodologías ágiles de desarrollo, y programación de entornos en IoT, en cuanto a la pertinencia, idoneidad y completitud del patrón arquitectónico de ubicación del procesamiento en ecosistemas IoT.

5.1.2. Objetivo de la investigación

El objetivo de la investigación es: realizar la evaluación inicial del patrón con el fin de encontrar oportunidades de mejora y generar una nueva iteración.

5.1.3. Preparación de materiales y procedimientos a seguir por parte del grupo de investigación

Los elementos, procedimientos y técnicas a emplear en la ejecución del grupo focal son: (i) protocolo del grupo focal, (ii) instrumentos y métodos empleados, (iii) definición de métodos de captura y registro de información y (iv) definición de los métodos de análisis de información obtenida en el debate.

5.1.4. Protocolo del grupo focal

En la Tabla 14 se presentan los elementos que componen el protocolo utilizado en el grupo focal.

Tabla 14. Protocolo del grupo focal

Número	Elemento	Descripción
1	Agenda de trabajo	Este documento describe las actividades que realizara cada participante durante la realización del grupo focal.
2	Cuestionario	Este documento contiene las preguntas de pertinencia, idoneidad y completitud para evaluar la propuesta en el grupo focal.
3	Estructura de protocolo	Este documento muestra el protocolo de aplicación del grupo focal.
4	Propuesta a evaluar	Este documento contiene la propuesta del patrón a evaluar.

5.1.5. Elementos necesarios para llevar a cabo el grupo focal

En la Tabla 15 se presentan los elementos utilizados para realizar el grupo focal.

Tabla 15. Elementos a tener en cuenta en la realización del Grupo Focal

Elemento	Descripción
Fecha de realización	10 de agosto de 2022.
Hora de inicio	11:00 a.m.
Duración	1 hora y 30 minutos.
Lugar	Sala virtual usando la plataforma MEET.
Tema a tratar	Conocer la opinión y percepción de expertos en IoT, metodologías ágiles de desarrollo, y programación de entornos en IoT en cuanto a la pertinencia, idoneidad y completitud del patrón arquitectónico de ubicación del procesamiento en ecosistemas IoT.
Moderador	Camilo Enrique Romero Parra.
Supervisor	Miguel Ángel Niño Zambrano.
Relator	Camilo Enrique Romero Parra.
Objetivo de investigación	Realizar la evaluación inicial del patrón con el fin de encontrar oportunidades de mejora y generar una nueva iteración.

5.1.6. Métodos de captura y registro de información

Para la realización del grupo focal se contó con la participación de un relator, encargado de tomar nota de los comentarios realizados por los participantes.

También se entregó a cada participante un cuestionario y se llevó registro audiovisual de las reuniones con cada participante.

5.1.7. Métodos de análisis de la información

Al finalizar el grupo focal se analizó la información recopilada en los cuestionarios y se realizó un análisis cualitativo con las observaciones y oportunidades de mejora registradas.

5.2. Diseño del grupo de discusión (Reclutamiento)

A continuación, se definen las estrategias de selección de los participantes del grupo focal.

5.2.1. Perfil del participante

Profesional en ingeniería en áreas afines a la computación, con estudios de maestría o doctorado, con conocimientos en el Internet de las Cosas o metodologías ágiles de desarrollo.

5.2.2. Identificación de los participantes

Teniendo en cuenta los criterios descritos anteriormente, se identificaron los posibles participantes al grupo focal. El grupo estuvo conformado por profesionales en IoT, metodologías ágiles de desarrollo, y programación de entornos en IoT. En la Tabla 16 se presenta la descripción del perfil profesional de los participantes.

Tabla 16. Perfil profesional de los participantes en el grupo focal

Id.	Estudios	Experiencia
PFG1	MSc en computación.	Investigador con enfoque en Entornos de desarrollo en IoT y Web semántica de las cosas
PFG2	MSc en ciencias de la computación, PhD. en ciencias de la computación.	Investigador con enfoque en metodologías ágiles de desarrollo
PFG3	MSc en ingeniería de sistemas, PhD. en ingeniería de sistemas.	Investigador con enfoque en Ingeniería de software y ambientes IoT

5.2.3. Conducción de la sección de debate

La sesión tiene una duración de una hora y treinta minutos, y es coordinada por un moderador que sigue el orden y la secuencia presentada en la Tabla 17.

Tabla 17. Organización del grupo focal

Numero	Descripción
1	Bienvenida a los participantes.
2	Presentación del equipo de trabajo, objetivos del grupo focal y objetivos de investigación.
4	Presentación del patrón arquitectónico de ubicación del procesamiento en ecosistemas IoT.
5	Preguntas de los participantes.
6	Retroalimentación por parte de los invitados.
7	Agradecimientos a los participantes.
8	Finalización del grupo focal.

Utilizando los métodos de captura y registro de información definidos en la fase 5.1.6, se realiza la interpretación de la información. Adicionalmente, se solicita a los expertos completar la encuesta proporcionada, La cual tiene como objetivo conocer la opinión de los expertos frente al patrón. Las preguntas contenidas en la encuesta se diseñan de tal manera que sus opciones de respuesta se limiten a las siguientes opciones: 1. Si, 2. No, 3. Tal vez.

Las preguntas buscan evaluar la pertinencia, idoneidad y completitud del patrón, minimizando el tiempo de respuesta y facilitando en el análisis de los resultados. Adicionalmente, las preguntas de completitud buscan conocer la opinión de los encuestados sobre el patrón y proponer mejoras. En la Tabla 18 se presentan las preguntas y el aspecto a evaluar.

Tabla 18. Cuestionario de evaluación usado en el grupo focal

Evaluación	Id.	Pregunta
Pertinencia	P1	¿Considera usted que la estructura propuesta para el patrón es adecuada para guiar el despliegue de una aplicación IoT?
	P2	¿Considera que los pasos del patrón son adecuados para guiar la elección de la ubicación del procesamiento en una aplicación IoT?
	P3	¿Considera que el patrón propuesto permite apoyar la decisión de donde desplegar el procesamiento de una aplicación IoT?
Idoneidad	P4	¿Consideraría usted la posibilidad de utilizar el patrón propuesto en un proyecto de despliegue real de una aplicación IoT?
Completitud	P5	¿Qué conceptos o elementos adicionales a los propuestos, a su juicio deben incluirse para mejorar el patrón propuesto?
	P6	¿Qué desventajas o inconvenientes a su juicio, tiene la solución propuesta?

5.2.4. Análisis de la información y reporte de resultados

Al finalizar el grupo focal, se analizan los comentarios de los participantes, así como las intervenciones realizadas en la sesión de discusión de la propuesta y de los cuestionarios diligenciados. A continuación, se presentan las actividades realizadas para analizar la información obtenida en el grupo focal. En el Anexo 2 se encuentran las respuestas de los participantes.

5.2.4.1. Análisis de las preguntas cerradas

Para las preguntas P1 a P4, se realizó el conteo de las respuestas de cada participante. En la Tabla 19 se presenta dicho conteo.

Tabla 19. Conteo de respuestas a preguntas P1 a P4

Id.	Respuesta		
	Si	No	Tal vez
P1	2	0	1
P2	2	0	1
P3	2	0	1
P4	2	0	1

A continuación, se presenta el análisis de las respuestas a las preguntas cerradas de acuerdo con el aspecto que evaluaron, pertinencia e idoneidad:

Pertinencia:

En P1 se les pregunto a los asistentes si consideraban que la estructura propuesta para el patrón es adecuada para guía el despliegue de una aplicación IoT. Dos participantes respondieron con la opción SI y uno de ellos respondió con la opción Tal vez. Al reflexionar sobre la repuesta “Tal vez” dada por el experto, se concluye que la incertidumbre, respecto al uso de dispositivos inteligentes y la experticia del usuario en técnicas de inteligencia artificial, puede jugar un papel determinante en el proceso de despliegue de recursos IoT.

En P2 se les pregunto a los asistentes si consideraban que los pasos del patrón son adecuados para guiar la elección de la ubicación del procesamiento en una aplicación IoT. Dos participantes respondieron con la opción SI y uno de ellos respondió con la opción Tal vez. Al reflexionar sobre la repuesta “Tal vez” dada por el experto, se concluye que junto a los pasos que componen el patrón podría incluirse un sistema que asesore al usuario sobre la ubicación más adecuada para su despliegue a partir de las condiciones particulares de la aplicación.

En P3 se les pregunto a los asistentes si consideraban que el patrón propuesto permite apoyar la decisión de donde desplegar el procesamiento de una aplicación IoT. Dos participantes respondieron con la opción SI y uno de ellos respondió con la opción Tal vez. Al reflexionar sobre la repuesta “Tal vez” dada por el experto, se

concluye que una oportunidad de mejora podría ser apoyar al usuario con una guía que le permita descartar las ubicaciones que no cumplan con sus requerimientos.

Idoneidad:

En P4 se les pregunto a los asistentes si considerarían la posibilidad de utilizar el patrón propuesto en un proyecto de despliegue real de una aplicación IoT. Dos participantes respondieron con la opción SI y uno de ellos respondió con la opción Tal vez. Al reflexionar sobre la respuesta “Tal vez” dada por el experto, se concluye que en su estado actual el patrón no permite guiar la ubicación del procesamiento de manera satisfactoria, debido a las condiciones de conocimientos previos de inteligencia artificial por parte del desarrollador de la aplicación.

Respecto a las respuestas obtenidas se concluye que el patrón es parcialmente pertinente e idóneo, y se resaltan las sugerencias de guiar al usuario con un sistema experto que le permita ubicar el procesamiento de la aplicación sin requerir de un conocimiento experto previo en inteligencia artificial.

5.2.4.2. Análisis de las preguntas abiertas

Las preguntas P5 y P6 evalúan la completitud del patrón y permiten a los asistentes proponer ajustes al modelo y realizar comentarios adicionales. A continuación, en la Tabla 20 se presentan las respuestas de cada uno de los participantes.

Tabla 20. Respuestas a las preguntas abiertas

Pregunta	Participante	Respuesta
P5	PGF1	Incluir elementos que permitan reducir subjetividad con respecto a la experiencia del desarrollado.
	PGF2	Sería interesante plantearse la generalidad del patrón, medirla en el estudio de caso y quizá establecer el límite del mismo, asimismo, dejar clara la manera en la cual el patrón establece o tiene en cuenta la conexión con otros sistemas ciber físicos a través del paradigma IoT, es decir, cómo se interactúa con otros dispositivos así como la seguridad, en especial, porque el quantum era (era cuántica) es un presente, sería interesante plantearse seguir este trabajo a futuro involucrando estos aspectos. Muy buen trabajo, felicidades.
	PGF3	Considerar el principio de incertidumbre que asiste el proceso de despliegue de recursos IT/ OT/ Comm, dado que se fundamentan fuertemente en las condiciones del entorno físico y el ecosistema con el que se aprovisionará la solución IoT.

P6	PGF1	La solución deja mucho a la subjetividad del desarrollador y existe incertidumbre acerca de si las decisiones tomadas son las óptimas.
	PGF2	Ninguna.
	PGF3	Extensibilidad.

5.2.4.3. Información extraída de la relatoría

Junto con el cuestionario realizado, durante la sesión de debate los participantes expresaron opiniones y comentarios sobre el patrón y la forma de implementación. A continuación, se listan las oportunidades de mejora proporcionadas durante el debate que no se reflejan en las preguntas de completitud.

Plantear como trabajo futuro un sistema experto que guie la elección de la ubicación del procesamiento de algoritmos de inteligencia artificial en ecosistemas IoT, utilizando los recursos del ecosistema IoT en sus diferentes módulos para cumplir con los objetivos del usuario final.

Cambiar el enfoque del patrón para usuarios que no cuentan con los conocimientos previos en desarrollo de aplicaciones IoT e inteligencia artificial, por medio de pasos que guíen de manera específica la ubicación del procesamiento.

5.2.4.4. Acciones de mejora

Con base en los resultados del grupo focal, que incluyen el cuestionario y la sesión de debate, a continuación, en la Tabla 21 se describe el cambio realizado por cada comentario o en su defecto la justificación para no realizar el cambio.

Tabla 21. Acciones de mejora definidas para el modelo

Comentario	Acción de Mejora / Justificación
Plantear la generalidad del patrón, medirla en el estudio de caso y quizá establecer el límite del mismo, asimismo, dejar clara la manera en la cual el patrón establece o tiene en cuenta la conexión con otros sistemas ciber físicos a través del paradigma IoT, es decir, cómo se interactúa con otros dispositivos, así como la seguridad, en especial, porque el quantum era (era cuántica) es un presente, sería interesante plantearse seguir este trabajo a futuro involucrando estos aspectos. Muy buen trabajo, felicidades.	Respecto a la sugerencia de establecer una conexión con sistemas ciber físicos, no se realizan cambios al patrón debido a que la sugerencia excede el alcance de los objetivos de la investigación. Respecto a la sugerencia de medir la generalidad del patrón, esta se especifica como parte de la intención y motivación del mismo, siendo un patrón que puede ser aplicado para todos los algoritmos de inteligencia artificial y todas las ubicaciones de un ecosistema IoT siempre que estas sean compatibles, junto con los objetivos del usuario.

<p>Considerar el principio de incertidumbre que asiste el proceso de despliegue de recursos IT/ OT/ Comm, dado que se fundamentan fuertemente en las condiciones del entorno físico y el ecosistema con el que se aprovisionará la solución IoT.</p>	<p>La sugerencia se plantea como trabajo futuro, pero no se le da solución debido a que implicaría realizar múltiples pruebas en las diferentes ubicaciones con distintos algoritmos para definir un rango de trabajo con distintos requisitos no funcionales, y como se pudo observar en la primera iteración del patrón, estas pruebas exceden el alcance de la investigación por disponibilidad de recursos y tiempo.</p>
--	--

5.2.4.5. Retroalimentación adicional

Gracias a la retroalimentación dada por los expertos fue posible realizar mejoras en el patrón propuesto que permitieron obtener la versión preliminar del mismo. Una vez aplicados los cambios sugeridos por el grupo focal, se decidió realizar una entrevista de forma individual con cada experto, con el fin de conocer su perspectiva frente a las modificaciones realizadas. En el Anexo 3 se encuentran las respuestas de los participantes a la entrevista de retroalimentación.

A continuación, en la Tabla 22, se presentan las preguntas realizadas a cada participante, las cuales son una evolución de las preguntas realizadas en el grupo focal.

Tabla 22. Cuestionario de evaluación de la entrevista

Evaluación	Id.	Pregunta
Pertinencia	P1	¿Considera usted que la estructura propuesta para el patrón es adecuada para guiar el despliegue de una aplicación IoT?
	P2	¿Considera que los pasos del patrón son adecuados para guiar la elección de la ubicación del procesamiento en una aplicación IoT?
	P3	¿Considera que el patrón propuesto permite apoyar la decisión de donde desplegar el procesamiento de una aplicación IoT?
Idoneidad	P4	¿Consideraría usted la posibilidad de utilizar el patrón propuesto en un proyecto de despliegue real de una aplicación IoT donde se requiera seleccionar la ubicación del procesamiento de IA?
Complejidad	P5	¿Qué conceptos o elementos adicionales a los propuestos, a su juicio deben incluirse para mejorar el patrón propuesto?
	P6	¿Qué desventajas o inconvenientes a su juicio, tiene la solución propuesta?

En la Tabla 23 se presentan las respuestas a las preguntas de pertinencia e idoneidad de cada uno de los participantes.

Tabla 23. Conteo de respuestas a preguntas P1 a P4

Id.	Respuesta		
	Si	No	Tal vez
P1	2	0	1
P2	3	0	0
P3	2	0	1
P4	2	0	1

Como se puede observar, los participantes consideran que los pasos del patrón son adecuados para guiar la elección de la ubicación del procesamiento, siendo esto una mejora con respecto a la versión anterior.

Adicionalmente, se presentan las respuestas de los participantes frente a las preguntas abiertas en la Tabla 24.

Tabla 24. Respuestas a las preguntas abiertas

Pregunta	Participante	Respuesta
P5	PGF1	Mayor nivel de granularidad en las preguntas asociadas.
	PGF2	Como trabajo futuro se podría plantear la posibilidad de dirigir una investigación encaminada al manejo de la subjetividad que surge en el momento de asignar las puntuaciones a cada uno de los apartados.
		Como trabajo futuro se podría plantear la necesidad de ampliar la propuesta para que pueda abarcar soluciones en las que el cómputo está distribuido. Podrían evaluar la posibilidad de incluir el factor precio en los criterios para la toma de decisiones. Podrían evaluar la posibilidad de incluir un apartado que permitiera evaluar las distintas opciones en cuanto a dispositivos y servicios en cada ubicación de la solución.
	PGF3	Ninguno, a mi juicio, pienso que es lo suficientemente completo como soportar el objetivo deseado en el proyecto.
P6	PGF1	Es necesario reflexionar que por ser un instrumento a priori, conlleva un alto nivel de experticia por parte del usuario. Es decir, a menor experticia mayor imprecisión y menor objetividad.
	PGF2	Solo la subjetividad que se puede presentar al momento de asignar las puntuaciones.
	PGF3	Ninguno.

Como se puede apreciar, los participantes del grupo focal manifiestan que la calidad del patrón a mejorado respecto a la primera versión presentada, eliminando en mayor medida la subjetividad de este.

Gracias a los comentarios recibidos, se realizaron las acciones de mejora que se presentan en la Tabla 25, las cuales condujeron a obtener la versión del patrón final que se detalló en la sección anterior.

Tabla 25. Acciones de mejora definidas para el modelo

Comentario	Acción de Mejora / Justificación
Podrían evaluar la posibilidad de incluir el factor precio en los criterios para la toma de decisiones.	Fue añadido el requisito no funcional de costo al patrón.
Es necesario reflexionar que por ser un instrumento a priori, conlleva un alto nivel de experticia por parte del usuario. Es decir, a menor experticia mayor imprecisión y menor objetividad.	No se realizó ninguna acción de mejora, debido a que el patrón en su estado actual no requiere que el desarrollador que lo utilice sea un experto (ya sea en inteligencia artificial o en IoT), pero si requiere de las nociones tecnológicas básicas o fundamentales para conocer los algoritmos que utilizara para dar solución a su historia de usuario o caso de uso, y de las herramientas hardware que tenga a su disposición, tanto sus fortalezas como limitaciones.

6. PRUEBA DE CONCEPTO

Las siguientes pruebas de concepto tienen como objetivo comprobar la utilidad del patrón titulado: “Definición del Punto de Despliegue de Aplicaciones Basadas en Inteligencia Artificial en la WoT”. En los dos casos planteados se precisa ubicar adecuadamente los algoritmos utilizados, de tal forma que se realice el despliegue en la ubicación que mejor resuelve la necesidad del usuario final. Para las pruebas de concepto se hace uso de un ambiente real.

6.1. Primer caso de estudio

En el caso de estudio propuesto se requiere realizar la detección de tumores cerebrales utilizando algoritmos de inteligencia artificial, teniendo como dato de entrada para el algoritmo imágenes de radiografías cerebrales suministradas por el usuario y como salida la clasificación dada por el algoritmo (libre de tumor o con tumor cerebral). La identificación debe realizarse en un periodo menor a 10 segundos ya que se utilizará para apoyar las labores de diagnóstico en un consultorio médico.

Para el entrenamiento del modelo se cuenta con un data set con 500 imágenes de radiografías cerebrales con tumores, y 500 sin tumores. Se cuenta con una conexión a internet inestable.

El desarrollador cuenta con un computador de escritorio. Además, tiene en mente implementar la detección de tumores utilizando una red neuronal convolucional.

Siguiendo los pasos del patrón propuesto (4.2), se realiza un entrenamiento previo del algoritmo en el computador obteniendo el modelo de la red neuronal que permitirá realizar el despliegue de la solución en la ubicación seleccionada. Luego, se realiza la evaluación de viabilidad en las tres posibles ubicaciones, como se muestra en la Sección 6.1.1. Posteriormente, se realiza la evaluación de cada ubicación y se determina cual será seleccionada para realizar el despliegue, como se muestra en la Sección 6.1.2. Finalmente en la Sección 6.1.3 se realiza el despliegue del algoritmo en la ubicación seleccionada.

6.1.1. Evaluación de viabilidad

A continuación, en la Tabla 26 se listan los datos requeridos para la aplicación, así como el lugar donde se alojan estos datos. Esta información se mantiene para las tres posibles ubicaciones.

Tabla 26. Descripción de los datos

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Haga un listado de los datos y los atributos de estos que requieren para entrenamiento y uso de la aplicación.	<ul style="list-style-type: none"> • Datos requeridos: Imágenes de radiografía cerebral. • Datos requeridos para entrenamiento: Imágenes e identificadores de las radiografías con y sin tumores con las que se va a entrenar el modelo. • Atributos requeridos: Nombre de la imagen con su estado (sin tumor o con tumor cerebral) y dimensiones de la imagen.
Defina donde están cada uno de esos datos.	<ul style="list-style-type: none"> • Los datos serán suministrados por el usuario y el algoritmo se encargará de dar el formato adecuado. • Los datos de entrenamiento se obtienen de una base de datos del centro médico.

La ubicación del Edge se descarta debido a que no se cuenta con dispositivos en el Edge con la capacidad de realizar el procesamiento en el tiempo requerido.

A continuación, en la Tabla 27, se realiza la evaluación de viabilidad en el Fog.

Tabla 27. Evaluación de la viabilidad del despliegue en el Fog

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> • Los datos serán proporcionados en esta ubicación por el usuario.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<ul style="list-style-type: none"> • Entorno de desarrollo integrado (IDE) para computador como VSCode o Pycharm (Para entrenamiento y ejecución del modelo). • Librerías: Pandas, Matplotlib, Numpy, Keras, Tensorflow.

¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> • Computador de escritorio, con un sistema operativo actualizado con parches de seguridad, con procesador Intel Core i7 de 8va generación, con tarjeta gráfica Nvidia GTX 1080, con 16GB de RAM.
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> • El dispositivo cuenta con la capacidad hardware necesarias para llevar a cabo el procesamiento y almacenamiento de los datos, así como la capacidad para conectarse a distintos dispositivos de la red. • Existen frameworks disponibles para desarrollar el algoritmo de procesamiento requerido.

Esta ubicación es viable, y por lo tanto pasa a la fase de calificación de requisitos no funcionales.

A continuación, en la Tabla 28, se realiza la evaluación de viabilidad en el Cloud.

Tabla 28. Evaluación de la viabilidad del despliegue en el Cloud

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> • Los datos pueden ser enviados al Cloud pues se cuenta con conectividad a internet.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus	<ul style="list-style-type: none"> • Entorno de programación Google Colab, Pycharm, VSCode, etc. (Para entrenamiento y ejecución del modelo).

similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<ul style="list-style-type: none"> • Librerías: Pandas, Matplotlib, Numpy, Keras, Tensorflow.
¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> • Depende de los servidores contratados.
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> • La ubicación cuenta con las capacidades necesarias para realizar el procesamiento. • Existen herramientas para llevar a cabo el procesamiento.

Esta ubicación es viable, y por lo tanto pasa a la fase de calificación de requisitos no funcionales.

De esta manera se sabe que las ubicaciones viables son el Fog y el Cloud.

6.1.2. Evaluación de las ubicaciones

Dentro de la evaluación de la importancia, se resalta que el montaje de un computador de escritorio con las características hardware descritas será más costoso que la contratación de un servicio de procesamiento alojado en el Cloud. Sin embargo, el tiempo de respuesta del Cloud, suponiendo una conexión a internet inestable obtendrá una baja puntuación en comparación al computador que no presenta latencia al no transferir datos entre ubicaciones.

La evaluación de la importancia se muestra a continuación en la Tabla 29:

Tabla 29. Evaluación de la importancia por requisito

Requisito	Importancia (0 menos importante y 5 más importante)
Tiempo de respuesta	5
Despliegue y actualización del modelo o del algoritmo	5
Adquisición de los datos	1
Procesamiento	5
Seguridad	4
Movilidad	0
Consumo energético	0
Costo	1

El resultado de la evaluación en el Fog del ejemplo se muestra a continuación en la Tabla 30:

Tabla 30. Ejemplo de evaluación de los requisitos no funcionales en el Fog

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
Tiempo de respuesta	5	5	25
Despliegue y actualización del modelo o del algoritmo	5	5	25
Adquisición de los datos	1	5	5
Procesamiento	5	3	15
Seguridad	4	3	12
Movilidad	0	0	0
Consumo energético	0	0	0
Costo	1	2	2
Total Acumulado	21		84
Calificación			4

La evaluación realizada en la Tabla 30 se muestra a continuación en la Figura 7:

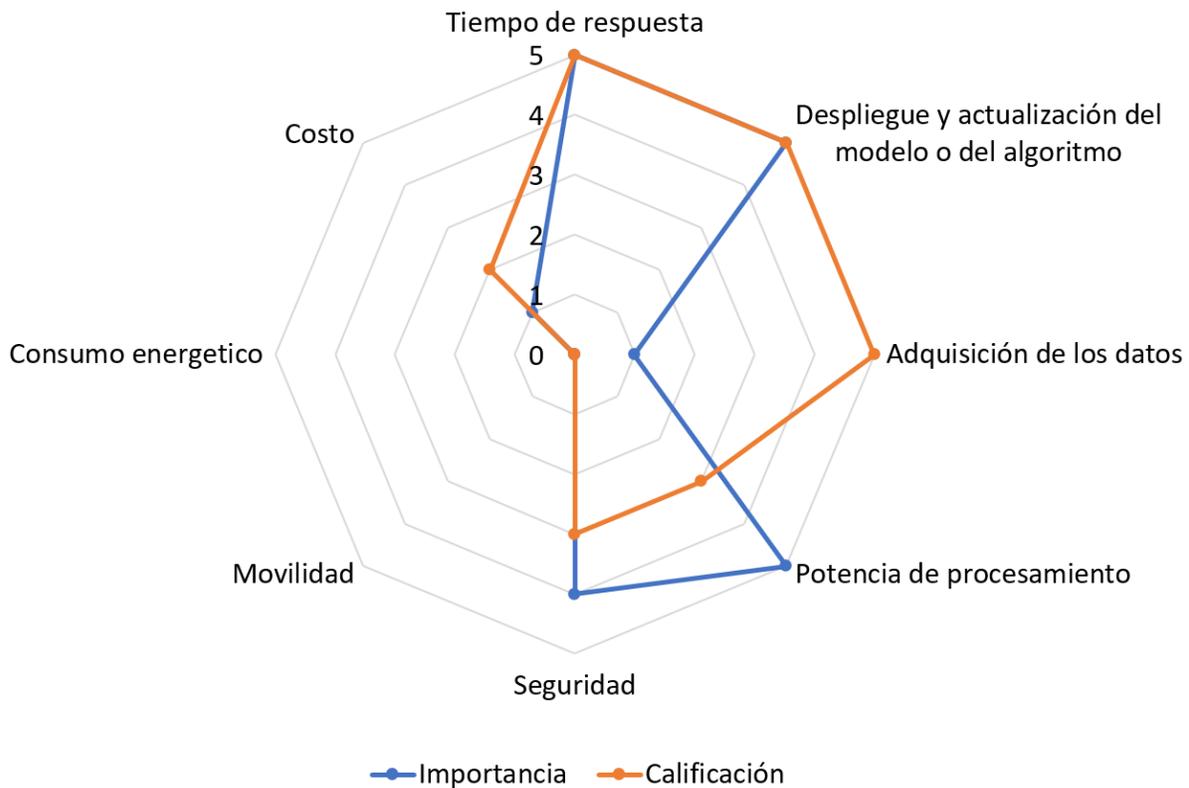


Figura 7. Gráfico de araña de calificación en el Fog. Imagen propia

Como se puede apreciar, la seguridad y la potencia de procesamiento no se ajustan con la importancia dada a estos requisitos no funcionales, siendo la potencia de procesamiento el requisito que más se aleja de la importancia establecida para el mismo.

El resultado de la evaluación en el Cloud del ejemplo se muestra a continuación en la Tabla 31:

Tabla 31. Ejemplo de evaluación de los requisitos no funcionales en el Cloud

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
Tiempo de respuesta	5	2	10
Despliegue y actualización del modelo o del algoritmo	5	5	25
Adquisición de los datos	1	5	5
Procesamiento	5	5	25
Seguridad	4	3	12
Movilidad	0	0	0
Consumo energético	0	0	0
Costo	1	4	4
Total Acumulado	21		81
Calificación			3,8571428

La evaluación realizada en la Tabla 31 se muestra a continuación en la Figura 8:

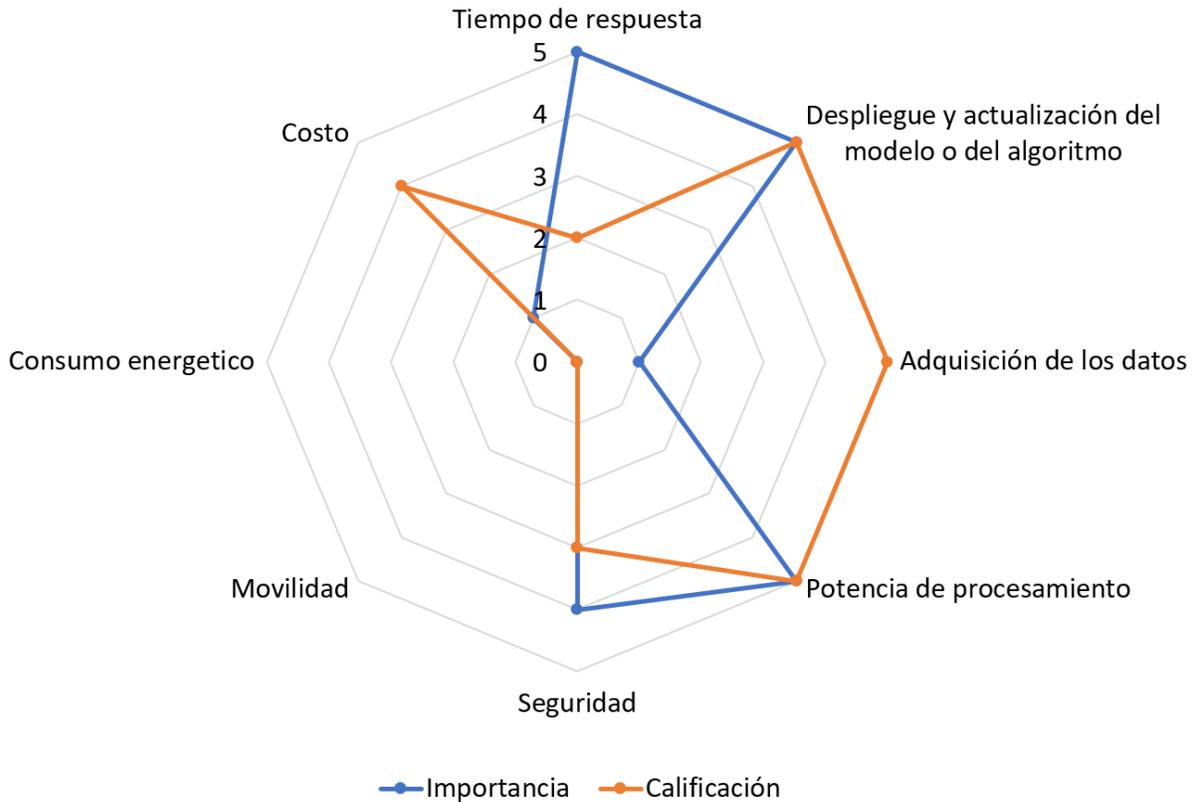


Figura 8. Gráfico de araña de calificación en el Cloud. Imagen propia

Como se puede apreciar, el tiempo de respuesta que se obtiene al evaluar la ubicación del Cloud no satisface las necesidades del usuario, de igual manera la seguridad no se ajusta con lo requerido.

A continuación, se presenta en la Figura 9 la comparativa entre las ubicaciones evaluadas.

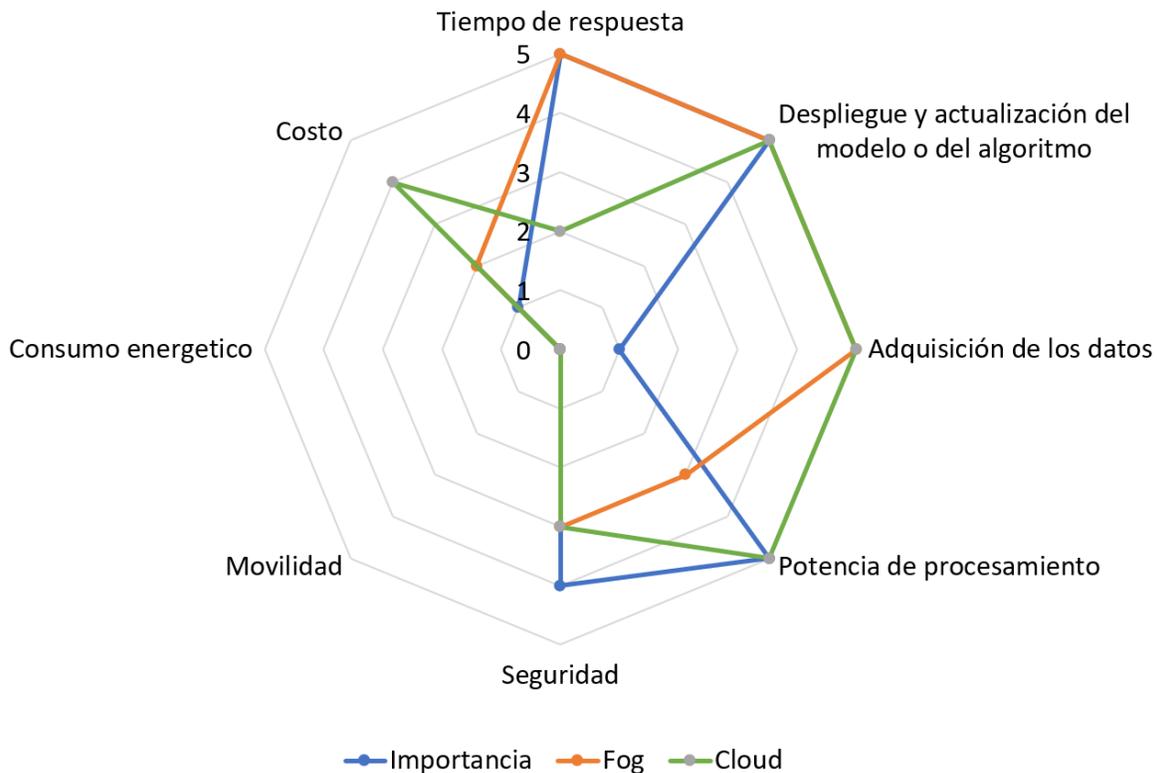


Figura 9. Gráfico de araña de comparativa de calificación. Imagen propia

Finalmente, obtenemos una calificación de 4 puntos para el Fog, superando la calificación de 3,85 puntos del Cloud, lo que indica que para la aplicación que se desea implementar el Fog será la mejor alternativa.

6.1.3. Despliegue

Para realizar el despliegue del algoritmo se utilizó el repositorio de Github, Brain-Tumer de Eran Feit [55].

El entrenamiento del modelo de la red neuronal tuvo una duración de 1 hora y 23 minutos, realizando el entrenamiento en el computador. En las Figura 10 y Figura 11 se muestra el resultado obtenido al desplegar el algoritmo en el computador (Fog), obteniendo la respuesta a la identificación solicitada en 3 segundos.

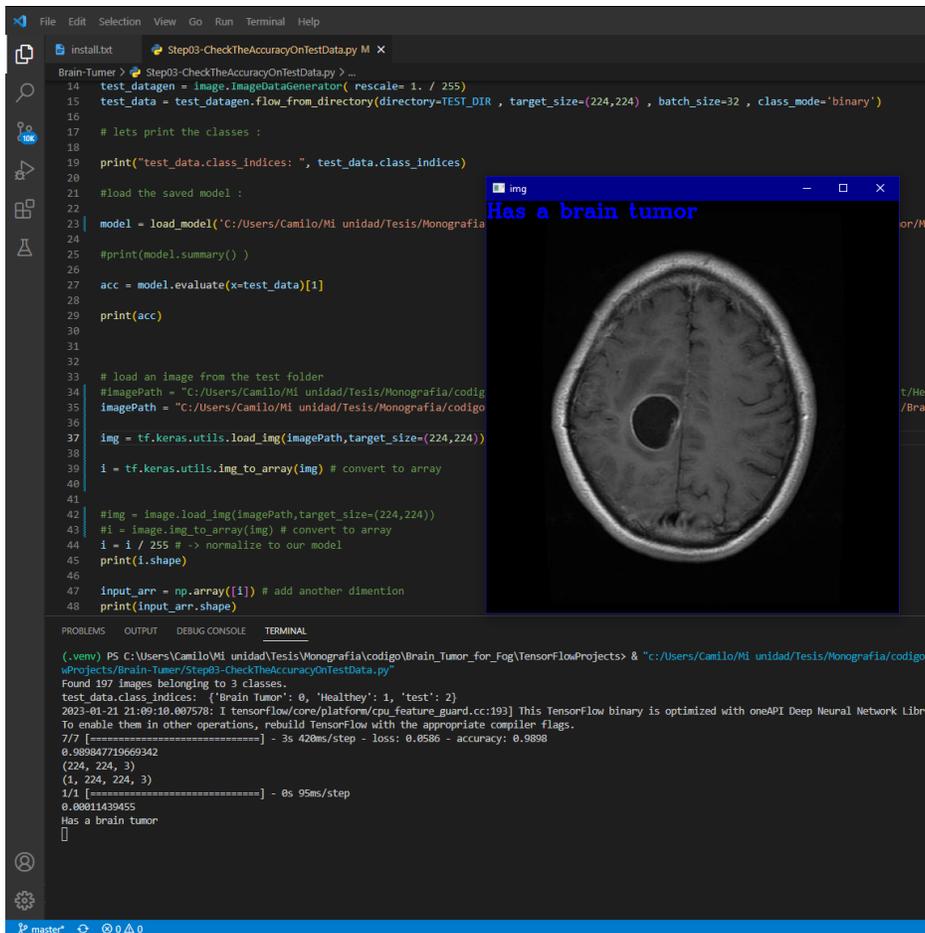


Figura 10. Detección de cáncer en cerebro enfermo. Imagen propia

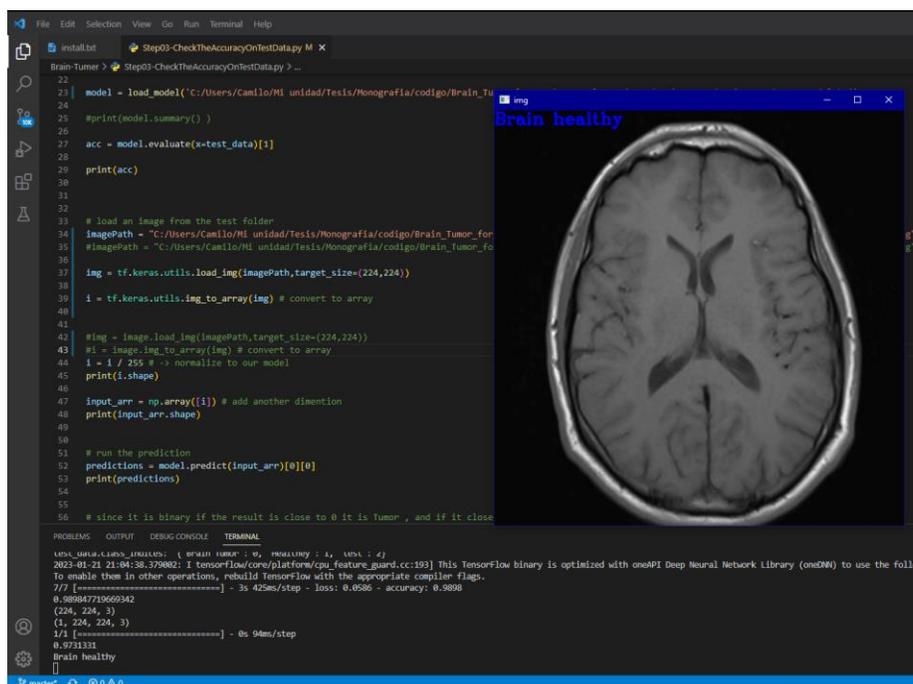


Figura 11. Detección de cáncer en cerebro sano. Imagen propia

Como se puede apreciar, el despliegue se realizó exitosamente cumpliendo con el tiempo requerido por el usuario.

6.2. Segundo caso de estudio

En el caso de estudio propuesto, una empresa de entretenimiento desea ofrecer un servicio de Deep Fake (modificación de expresiones faciales) utilizando las fotografías de sus clientes para crear clips de video en donde se imita el movimiento facial de un video de guía que puede ser proporcionado por el usuario o por la empresa. El video debe ser generado en un tiempo menor a 3 minutos.

El algoritmo utilizado se toma del repositorio que aloja los resultados obtenidos del estudio [56]. Para el entrenamiento del modelo se cuenta con una base de datos proporcionada por Jian Zhao et. al, en el estudio antes mencionado.

El desarrollador cuenta con un computador de escritorio. El algoritmo se basa en la transformación de movimiento para animación de imágenes con aprendizaje no supervisado.

Se realiza un entrenamiento previo del algoritmo en el Cloud, obteniendo su modelo. Luego, se realiza la evaluación de viabilidad en las tres posibles ubicaciones, como se muestra en la Sección 6.2.1.

6.2.1. Evaluación de viabilidad

A continuación, en la Tabla 32 se listan los datos requeridos para la aplicación, así como el lugar donde se alojan estos datos. Esta información se mantiene para las tres posibles ubicaciones.

Tabla 32. Descripción de los datos

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Haga un listado de los datos y los atributos de estos que requieren para entrenamiento y uso de la aplicación.	<ul style="list-style-type: none">• Datos requeridos: Fotografía del cliente.• Datos requeridos para entrenamiento: Imágenes e identificadores de rostros, junto con videos de personas demostrando expresiones faciales.• Atributos requeridos: Fotografía con sus dimensiones.
Defina donde están cada uno de esos datos.	<ul style="list-style-type: none">• Los datos serán suministrados por el cliente.• Los datos de entrenamiento se obtienen de una base de datos que proporciona el creador del algoritmo.

La ubicación del Edge se descarta debido a que no se cuenta con dispositivos que cuenten con la potencia de procesamiento necesaria para crear los videos solicitados, y de hacerlo, el tiempo de respuesta no sería aceptable para las necesidades del negocio.

A continuación, en la Tabla 33, se realiza la evaluación de viabilidad en el Fog.

Tabla 33. Evaluación de la viabilidad del despliegue en el Fog

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> Los datos serán proporcionados en esta ubicación por el usuario.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<ul style="list-style-type: none"> Entorno de desarrollo integrado (IDE) para computador como VSCode o Pycharm (Para entrenamiento y ejecución del modelo). Librerías: Torch, Imageio, Skimage, Matplotlib, Numpy.
¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> Computador de escritorio, con un sistema operativo actualizado con parches de seguridad, con procesador Intel Core i3 de 6ta generación, sin tarjeta gráfica, con 4GB de RAM.
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria, almacenamiento, red) que le permiten	<ul style="list-style-type: none"> El dispositivo cuenta con la capacidad hardware necesarias para llevar a cabo el procesamiento y almacenamiento de los datos.

usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> Existen frameworks disponibles para desarrollar el algoritmo de procesamiento requerido.
--	--

Esta ubicación es viable, y por lo tanto pasa a la fase de calificación de requisitos no funcionales.

A continuación, en la Tabla 34 se realiza la evaluación de viabilidad en el Cloud.

Tabla 34. Evaluación de la viabilidad del despliegue en el Cloud

¿Los datos requeridos para proveer el servicio se encuentran disponibles en la ubicación o pueden ser adquiridos desde está?	
Asegúrese de que cada uno de los datos están disponibles desde la ubicación que se está evaluando como opción de despliegue.	<ul style="list-style-type: none"> Los datos pueden ser enviados al Cloud pues se cuenta con conectividad a internet.
¿Existen herramientas para el desarrollo del procesamiento y análisis con inteligencia artificial, como frameworks o librerías con las implementaciones requeridas según el enfoque (machine learning, neural networks, expert systems, fuzzy logic o metaheuristics) de IA seleccionado?	
Basada en el tipo de implementación de IA, liste los frameworks o librerías que soportaran el despliegue de la solución, y que están disponibles en esta ubicación, que además cuenten con sus similares en el entorno de entrenamiento, desarrollo o modelado que se utilizara.	<ul style="list-style-type: none"> Entorno de programación Google Colab, Pycharm, VSCode, etc. (Para entrenamiento y ejecución del modelo). Librerías: Torch, Imageio, Skimage, Matplotlib, Numpy.
¿El dispositivo cuenta con las capacidades hardware para realizar el procesamiento (procesador y RAM), almacenar los datos y conectarse con otras fuentes de datos si esto se requiere (red)?	
Haga un inventario de los dispositivos disponibles en esta ubicación que serán utilizados como puntos de despliegue de la implementación basada en IA. Si la empresa no cuenta con los dispositivos y los va a adquirir tendrá que hacer una revisión de cuáles son los que le brindan las mejores capacidades (procesamiento, memoria, almacenamiento, consumo, etc.) con las restricciones presupuestales de la empresa o proyecto.	<ul style="list-style-type: none"> Depende de los servidores contratados. Para el caso de la implementación se utiliza un procesamiento por GPU con una tarjeta Tesla T4.
Asegúrese de que cada uno de estos dispositivos tiene las capacidades (procesamiento, memoria,	<ul style="list-style-type: none"> La ubicación cuenta con las capacidades necesarias para realizar el procesamiento.

almacenamiento, red) que le permiten usar los frameworks y librerías disponibles para el despliegue de la aplicación.	<ul style="list-style-type: none"> Existen herramientas para llevar a cabo el procesamiento.
---	---

Esta ubicación es viable, y por lo tanto pasa a la fase de calificación de requisitos no funcionales.

De esta manera se sabe que las ubicaciones viables son el Fog y el Cloud.

6.2.2. Evaluación de las ubicaciones

Dentro de la evaluación de la importancia, se resalta que el computador de escritorio con las características hardware descritas cuenta con las herramientas para realizar el procesamiento de forma limitada, con un tiempo de respuesta mayor al obtenido en el Cloud.

La evaluación de la importancia se muestra a continuación en la Tabla 35:

Tabla 35. Evaluación de la importancia por requisito

Requisito	Importancia (0 menos importante y 5 más importante)
Tiempo de respuesta	5
Despliegue y actualización del modelo o del algoritmo	5
Adquisición de los datos	3
Procesamiento	5
Seguridad	1
Movilidad	0
Consumo energético	0
Costo	3

El resultado de la evaluación en el Fog del ejemplo se muestra a continuación en la Tabla 36:

Tabla 36. Ejemplo de evaluación de los requisitos no funcionales en el Fog

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
Tiempo de respuesta	5	3	15
Despliegue y actualización del modelo o del algoritmo	5	5	25
Adquisición de los datos	3	5	15
Procesamiento	5	3	15
Seguridad	1	3	3
Movilidad	0	0	0

Patrón para Identificar en la Web de las Cosas los Puntos de Despliegue de Algoritmos de Inteligencia Computacional

Consumo energético	0	3	0
Costo	3	5	15
Total Acumulado	22		88
Calificación			4

La evaluación realizada en la Tabla 36 se muestra a continuación en la Figura 12:

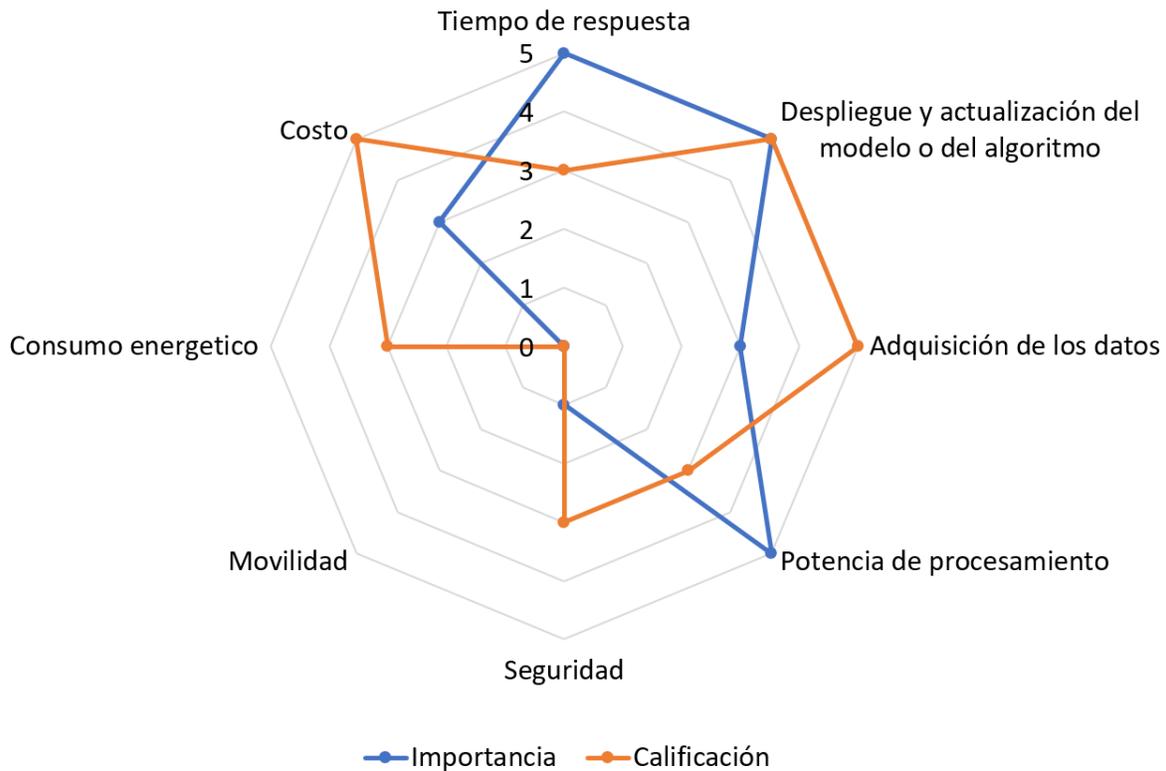


Figura 12. Gráfico de araña de calificación en el Fog. Imagen propia

Como se puede apreciar, la calificación obtenida en el Fog no se ajusta con el tiempo de respuesta y por la potencia de procesamiento requerida.

El resultado de la evaluación en el Cloud del ejemplo se muestra a continuación en la Tabla 37:

Tabla 37. Ejemplo de evaluación de los requisitos no funcionales en el Cloud

Requisito no funcional con su pregunta asociada	Importancia	Calificación de la ubicación (Edge, Fog o Cloud) de 0 a 5	Calificación Ponderada
Tiempo de respuesta	5	5	25
Despliegue y actualización del modelo o del algoritmo	5	5	25
Adquisición de los datos	3	4	12
Procesamiento	5	5	25
Seguridad	1	3	3

Patrón para Identificar en la Web de las Cosas los Puntos de Despliegue de Algoritmos de Inteligencia Computacional

Movilidad	0	0	0
Consumo energético	0	0	0
Costo	3	3	9
Total Acumulado	22		99
Calificación			4,5

La evaluación realizada en la Tabla 37 se muestra a continuación en la Figura 13:

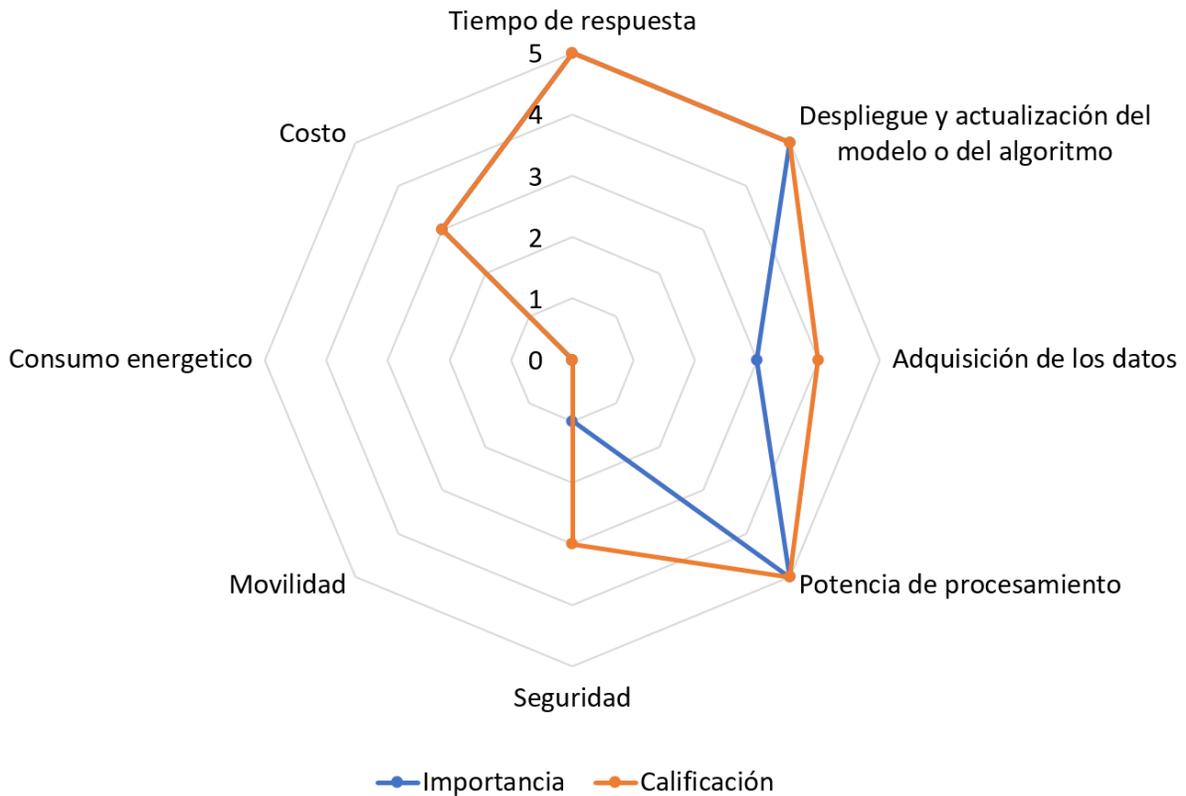


Figura 13. Gráfico de araña de calificación en el Cloud. Imagen propia

Como se puede apreciar, todos los requisitos no funcionales cuentan con una calificación que se ajusta a la importancia requerida.

A continuación, se presenta en la Figura 14 la comparativa entre las ubicaciones evaluadas.

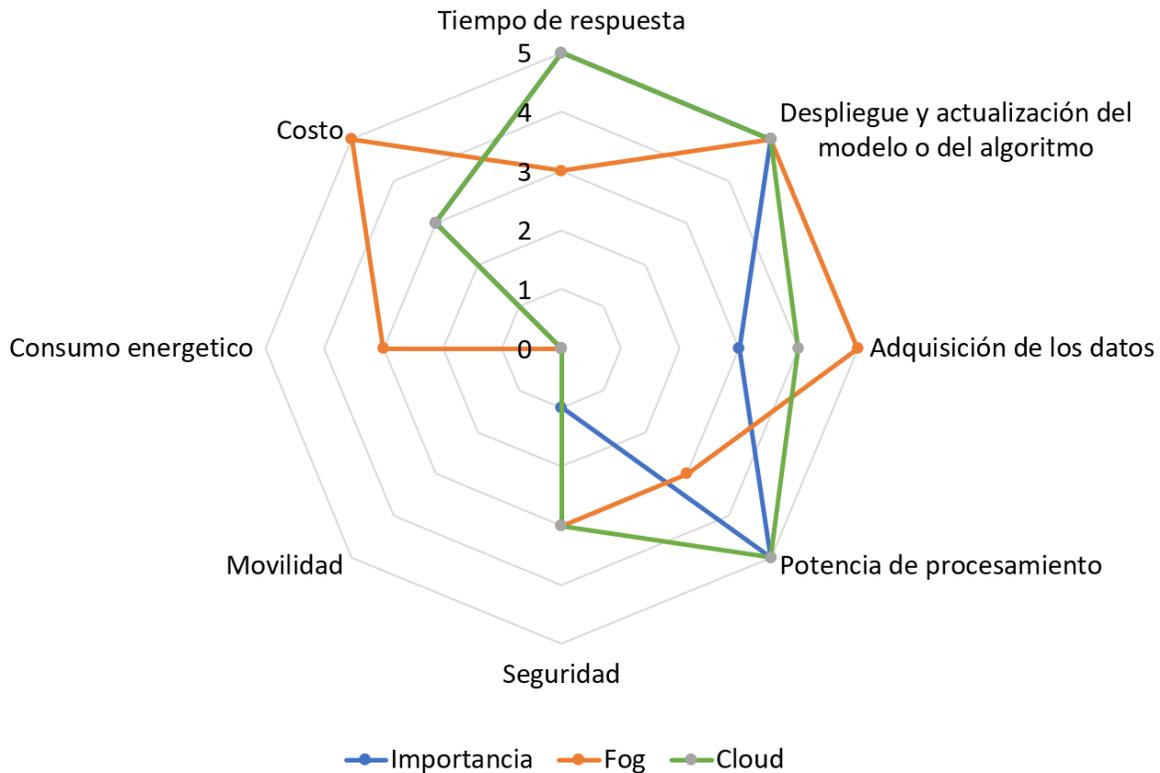


Figura 14. Gráfico de araña de comparativa de calificación. Imagen propia

Finalmente, obtenemos una calificación de 4 puntos para el Fog, y una calificación de 4,5 puntos del Cloud, lo que indica que para la aplicación que se desea implementar el Cloud será la mejor alternativa.

6.2.3. Despliegue

Para realizar el despliegue del algoritmo se utilizó el repositorio de Github, Thin-Plate-Spline-Motion-Model de Jian Zhao [56].

El entrenamiento del modelo tuvo una duración de 21 minutos, realizando el entrenamiento en el Cloud. En la Figura 15 se muestra el resultado obtenido al desplegar el algoritmo en el Cloud, obteniendo el video solicitado en 1 minuto.

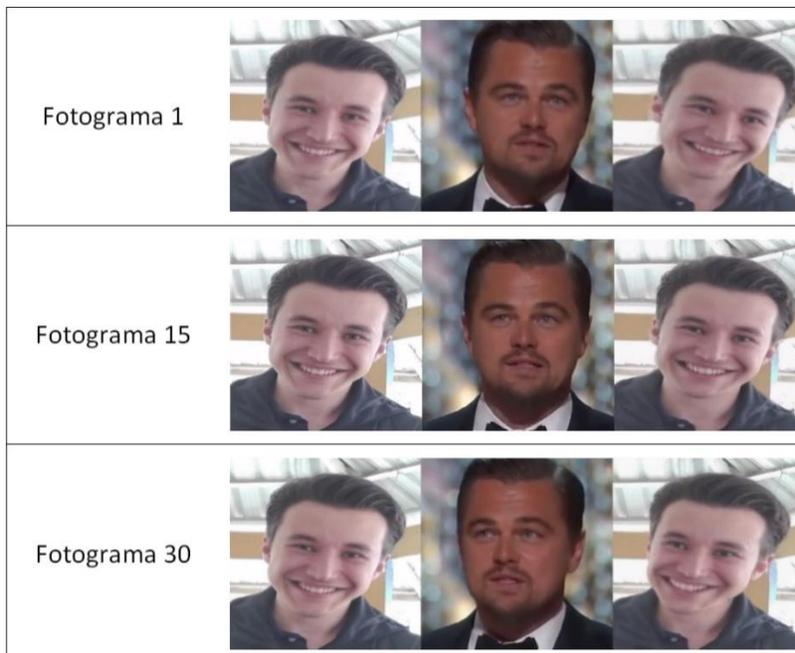


Figura 15. Fotogramas del video generado por el algoritmo. Imagen propia

Como se puede apreciar, el despliegue se realizó exitosamente cumpliendo con el tiempo requerido por el usuario.

7. CONCLUSIONES Y TRABAJOS FUTUROS

A continuación, se presentan las conclusiones a las cuales fue posible llegar con la investigación realizada:

- El presente estudio buscó responder la pregunta de investigación: ¿Cómo determinar la ubicación (edge, fog, o cloud) de los algoritmos de inteligencia computacional en un IOEoWoT, con el fin de obtener el mejor tiempo de respuesta en el consumo de los servicios soportados? Una vez realizada la investigación se logró proponer la primera versión de un patrón arquitectónico que orienta la toma de la decisión de la mejor ubicación de una aplicación basada en inteligencia artificial; dicho patrón incluye un ejemplo de uso de este en una aplicación de selección de piezas defectuosas en imágenes tomadas en una línea de producción. Además, se muestran dos pruebas de concepto, una para detección de cáncer basado en imágenes, y otra para generación de video con base en imágenes, dando como resultado que según los requisitos no funcionales de cada aplicación la primera se debe ubicar en el Fog y la segunda en el Cloud.
- Durante el desarrollo del patrón fue posible entender que la selección de la ubicación del procesamiento no se limita a las capacidades hardware con las que cuenta una ubicación, sino que se hace necesario analizar las necesidades del usuario final y las limitaciones que se puedan presentar respecto al tiempo de respuesta, despliegue y actualización del modelo o del algoritmo, adquisición de los datos, procesamiento, seguridad, movilidad, consumo energético y costo.
- Dentro del desarrollo del patrón se tomó la decisión de no solo abarcar algoritmos de inteligencia computacional (neural networks, fuzzy logic y metaheuristics), sino que también se tomaron en cuenta todos los enfoques que se incluyen dentro de la inteligencia artificial (machine learning, neural networks, expert systems, fuzzy logic y metaheuristics), aumentando su utilidad.
- Para evaluar la pertinencia de la investigación, se realizó una revisión sistemática de la literatura, gracias a la cual se evidencia la falta de estudios que realicen comparativas de procesamiento entre distintas ubicaciones de un ecosistema de objetos inteligentes utilizando algoritmos de inteligencia computacional.
- El patrón titulado: “Definición del Punto de Despliegue de Aplicaciones Basadas en Inteligencia Artificial en la WoT”, que se propone en este trabajo de grado, ha demostrado ser de utilidad para guiar la elección del procesamiento en ecosistemas inteligentes cuando se hace uso de algoritmos de inteligencia artificial, como se pudo evidenciar en las pruebas de concepto realizadas. El

patrón fue desarrollado utilizando el patrón de investigación iterativa propuesto por Pratt.

- Como parte de la evaluación del patrón se realizó un grupo focal compuesto por expertos en el área de IoT y metodologías ágiles de desarrollo. Como principal limitación del patrón se identificó la incertidumbre ante la variedad de dispositivos hardware, específicamente en las ubicaciones del Edge y el Fog, y su compatibilidad con algoritmos de inteligencia artificial. Gracias a los comentarios realizados por los expertos, se obtuvo la versión definitiva del patrón, la cual fue presentada nuevamente a los expertos a través de una entrevista individual, en la cual los expertos mencionaron que el resultado final corrige varios de los puntos de mejora sugeridos en la presentación inicial.
- Los requisitos no funcionales que tiene en cuenta el patrón deberán ser evaluados respecto a su calidad previamente, dado que esta evaluación no forma parte del patrón propuesto.

Basado en las limitaciones que se identificaron del patrón propuesto, a continuación se presentan los trabajos futuros que el grupo de investigación espera desarrollar en el corto plazo:

- Realizar un análisis comparativo entre los resultados del procesamiento obtenidos con distintos dispositivos inteligentes ubicados en distintas partes de un ecosistema inteligente al implementar algoritmos de inteligencia computacional, guiando la ubicación con el patrón propuesto.
- Estudiar las limitaciones que puede presentar el patrón propuesto al implementar múltiples algoritmos o múltiples funcionalidades dentro de una misma solución.
- Agregar otros requisitos no funcionales dentro de una segunda versión del patrón tales como: escalabilidad, interoperabilidad, portabilidad, otros aspectos de mantenibilidad, actuación (el sistema debe manejar el número requerido de usuarios sin que se degrade el rendimiento), disponibilidad y compatibilidad. También incluir el análisis del contexto semántico en la IoT.

8. BIBLIOGRAFÍA

- [1] P. Sethi y S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications", *J. Electr. Comput. Eng.*, vol. 2017, p. 9324035, 2017, doi: 10.1155/2017/9324035.
- [2] G. Bai, L. Yan, L. Gu, Y. Guo, y X. Chen, "Context-aware usage control for web of things", *Secur. Commun. Networks*, vol. 7, núm. 12, pp. 2696–2712, 2014, doi: 10.1002/sec.424.
- [3] A. Wagner, J. L. V. Barbosa, y D. N. F. Barbosa, "A model for profile management applied to ubiquitous learning environments", *Expert Syst. Appl.*, vol. 41, núm. 4 PART 2, pp. 2023–2034, mar. 2014, doi: 10.1016/j.eswa.2013.08.098.
- [4] L. Yao, "A Propagation Model for Integrating Web of Things and Social Networks", en *Service-Oriented Computing - ICSOC 2011 Workshops*, 2012, pp. 233–238.
- [5] M. Niño Zambrano, "Interacción Semántica de Objetos en la Web de las Cosas", 2013.
- [6] B. Varghese y R. Buyya, "Next generation cloud computing: New trends and research directions", *Futur. Gener. Comput. Syst.*, vol. 79, pp. 849–861, feb. 2018, doi: 10.1016/j.future.2017.09.020.
- [7] A. Jaddoa, G. Sakellari, E. Panaousis, G. Loukas, y P. G. Sarigiannidis, "Dynamic decision support for resource offloading in heterogeneous Internet of Things environments", *Simul. Model. Pract. Theory*, vol. 101, p. 102019, may 2020, doi: 10.1016/j.simpat.2019.102019.
- [8] B. Alturki, S. Reiff-Marganiec, C. Perera, y S. De, "Exploring the Effectiveness of Service Decomposition in Fog Computing Architecture for the Internet of Things", *IEEE Trans. Sustain. Comput.*, p. 1, 2019, doi: 10.1109/TSUSC.2019.2907405.
- [9] G. M. S. Rahman, M. Peng, S. Yan, y T. Dang, "Learning Based Joint Cache and Power Allocation in Fog Radio Access Networks", *IEEE Trans. Veh. Technol.*, vol. 69, núm. 4, pp. 4401–4411, 2020, doi: 10.1109/TVT.2020.2975849.
- [10] N. Abbas, Y. Zhang, A. Taherkordi, y T. Skeie, "Mobile Edge Computing: A Survey", *IEEE Internet of Things Journal*, vol. 5, núm. 1. Institute of Electrical and Electronics Engineers Inc., pp. 450–465, febrero de 2018. doi: 10.1109/JIOT.2017.2750180.
- [11] Z. Mahmood y M. Ramachandran, "Fog Computing: Concepts, Principles and Related Paradigms", en *Fog Computing: Concepts, Frameworks and Technologies*, Z. Mahmood, Ed. Cham: Springer International Publishing, 2018, pp. 3–21. doi: 10.1007/978-3-319-94890-4_1.
- [12] C. Mechalikh, H. Taktak, y F. Moussa, "A Scalable and Adaptive Tasks Orchestration Platform for IoT", en *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)*, 2019, pp. 1557–1563. doi: 10.1109/IWCMC.2019.8766744.

- [13] K. BA y S. Charters, “Guidelines for performing Systematic Literature Reviews in Software Engineering”, vol. 2, 2007.
- [14] K. S. Pratt, “Design Patterns for Research Methods: Iterative Field Research”, en *AAAI Spring Symp. Exp. Des. Real*, 2009, núm. 1994, pp. 1–7.
- [15] P. Avgeriou y U. Zdun, “Architectural Patterns Revisited - A Pattern Language.”, 2005, vol. 81, pp. 431–470.
- [16] E. Gamma, R. Helm, R. Johnson, y J. Vlissides, “Design Patterns: Abstraction and Reuse of Object-Oriented Design”, 1993, pp. 406–431. doi: 10.1007/978-3-642-48354-7_15.
- [17] O. Vermesan *et al.*, “Internet of Things Strategic Research Roadmap”, 2009.
- [18] N. B. Ruparelia, *Cloud computing*. Mit Press, 2016. [En línea]. Disponible en: https://books.google.com.co/books?hl=es&lr=&id=umIsDAAQBAJ&oi=fnd&pg=PR5&dq=Definition+of+Cloud+Computing&ots=k8BtuSpQ29&sig=FylvnPN0V2IO12qQIPRrSekcaMk&redir_esc=y#v=onepage&q=Definition+of+Cloud+Computing&f=false
- [19] A. Jain y P. Singhal, “Fog computing: Driving force behind the emergence of edge computing”, en *2016 International Conference System Modeling Advancement in Research Trends (SMART)*, nov. 2016, pp. 294–297. doi: 10.1109/SYSMART.2016.7894538.
- [20] P. Pedamkar, “IoT Ecosystem”. 2020. Consultado: el 26 de mayo de 2021. [En línea]. Disponible en: <https://www.educba.com/iot-ecosystem/>
- [21] J. McCarthy, “What is Artificial Intelligence?”, 2004.
- [22] E. Alpaydın, *Introduction to Machine Learning*. 2004.
- [23] P. Jackson, “Introduction to expert systems”, p. 246, 1986.
- [24] T. Hanne y R. Dornberger, “Computational Intelligence”, en *Computational Intelligence in Logistics and Supply Chain Management*, Cham: Springer International Publishing, 2017, pp. 13–41. doi: 10.1007/978-3-319-40722-7_2.
- [25] R. E. Uhrig, “Introduction to artificial neural networks”, *Proc. IECON '95 - 21st Annu. Conf. IEEE Ind. Electron.*, vol. 1, pp. 33–37, doi: 10.1109/IECON.1995.483329.
- [26] H. Takagi, “Introduction to Fuzzy Systems, Neural Networks, and Genetic Algorithms”, *Intell. Hybrid Syst.*, pp. 3–33, 1997, doi: 10.1007/978-1-4615-6191-0_1.
- [27] G. (Guanrong) Chen y T. T. Pham, “Introduction to fuzzy systems”, p. 315, 2006.
- [28] B. Chopard y M. Tomassini, “An Introduction to Metaheuristics for Optimization”, 2018, doi: 10.1007/978-3-319-93073-2.
- [29] H. Washizaki, S. Ogata, A. Hazeyama, T. Okubo, E. B. Fernandez, y N. Yoshioka, “Landscape of Architecture and Design Patterns for IoT Systems”, *IEEE Internet Things J.*, vol. 7, núm. 10, pp. 10091–10101, oct. 2020, doi: 10.1109/JIOT.2020.3003528.
- [30] B. Rababah, T. Alam, y R. Eskicioglu, “The Next Generation Internet of Things Architecture Towards Distributed Intelligence: Reviews, Applications, and Research Challenges”, *SSRN Electron. J.*, 2020, doi: 10.2139/ssrn.3640136.
- [31] M. Aazam, S. Zeadally, y K. A. Harras, “Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities”, *Futur. Gener. Comput. Syst.*, vol. 87, pp. 278–289, oct. 2018, doi:

- 10.1016/J.FUTURE.2018.04.057.
- [32] K. Petersen, S. Vakkalanka, y L. Kuzniarz, “Guidelines for conducting systematic mapping studies in software engineering: An update”, en *Information and Software Technology*, 2015, vol. 64, pp. 1–18. doi: 10.1016/j.infsof.2015.03.007.
- [33] A. Malekian Borujeni, M. Fathy, y N. Mozayani, “A hierarchical, scalable architecture for a real-time monitoring system for an electrocardiography, using context-aware computing”, *J. Biomed. Inform.*, vol. 96, p. 103251, ago. 2019, doi: 10.1016/J.JBI.2019.103251.
- [34] P. Y. Zhang, M. C. Zhou, y G. Fortino, “Security and trust issues in Fog computing: A survey”, *Futur. Gener. Comput. Syst.*, vol. 88, pp. 16–27, nov. 2018, doi: 10.1016/J.FUTURE.2018.05.008.
- [35] R. B. Almeida *et al.*, “A distributed event-driven architectural model based on situational awareness applied on internet of things”, *Inf. Softw. Technol.*, vol. 111, pp. 144–158, jul. 2019, doi: 10.1016/j.infsof.2019.04.001.
- [36] T. Zhang, Z. Shen, J. Jin, X. Zheng, A. Tagami, y X. Cao, “Achieving Democracy in Edge Intelligence: A Fog-Based Collaborative Learning Scheme”, *IEEE Internet Things J.*, vol. 8, núm. 4, pp. 2751–2761, feb. 2021, doi: 10.1109/JIOT.2020.3020911.
- [37] L. U. Khan *et al.*, “Federated Learning for Edge Networks: Resource Optimization and Incentive Mechanism”, *IEEE Commun. Mag.*, vol. 58, núm. 10, pp. 88–93, 2020, doi: 10.1109/MCOM.001.1900649.
- [38] J. An *et al.*, “EiF: Toward an Elastic IoT Fog Framework for AI Services”, *IEEE Commun. Mag.*, vol. 57, núm. 5, pp. 28–33, may 2019, doi: 10.1109/MCOM.2019.1800215.
- [39] Z. Lv, D. Chen, R. Lou, y Q. Wang, “Intelligent edge computing based on machine learning for smart city”, *Futur. Gener. Comput. Syst.*, vol. 115, pp. 90–99, feb. 2021, doi: 10.1016/j.future.2020.08.037.
- [40] J. Al-Jaroodi y N. Mohamed, “PsCPS: A distributed platform for cloud and fog integrated smart cyber-physical systems”, *IEEE Access*, vol. 6, pp. 41432–41449, jul. 2018, doi: 10.1109/ACCESS.2018.2856509.
- [41] D. Schäfer, “Elastic computation placement in edge-based environments”, 2019, Consultado: el 17 de abril de 2021. [En línea]. Disponible en: <https://madoc.bib.uni-mannheim.de/48832>
- [42] T. Leppänen y J. Riekkki, “Energy Efficient Opportunistic Edge Computing for the Internet of Things”, *Web Intell. Agent Syst.*, vol. 17, 2018, doi: 10.3233/WEB-190414.
- [43] W. Yáñez, R. Mahmud, R. Bahsoon, Y. Zhang, y R. Buyya, “Data Allocation Mechanism for Internet-of-Things Systems With Blockchain”, *IEEE Internet Things J.*, vol. 7, núm. 4, pp. 3509–3522, 2020, doi: 10.1109/JIOT.2020.2972776.
- [44] R. I. Ciobanu, C. Negru, F. Pop, C. Dobre, C. X. Mavromoustakis, y G. Mastorakis, “Drop computing: Ad-hoc dynamic collaborative computing”, *Futur. Gener. Comput. Syst.*, vol. 92, pp. 889–899, mar. 2019, doi: 10.1016/j.future.2017.11.044.
- [45] R. Calegari, G. Ciatto, E. Denti, y A. Omicini, “Engineering Micro-intelligence at the Edge of CPCS: Design Guidelines”, 2019, pp. 260–270. doi:

- 10.1007/978-3-030-34914-1_25.
- [46] B. D. Deebak, F. Al-Turjman, M. Aloqaily, y O. Alfandi, “IoT-BSFCAN: A smart context-aware system in IoT-Cloud using mobile-fogging”, *Futur. Gener. Comput. Syst.*, vol. 109, pp. 368–381, ago. 2020, doi: 10.1016/j.future.2020.03.050.
- [47] S. K. Sharma y X. Wang, “Live Data Analytics with Collaborative Edge and Cloud Processing in Wireless IoT Networks”, *IEEE Access*, vol. 5, pp. 4621–4635, 2017, doi: 10.1109/ACCESS.2017.2682640.
- [48] A. Dounis, “Artificial intelligence for energy conservation in buildings”, *Adv. Build. Energy Res.*, vol. 4, pp. 267–299, 2010, doi: 10.3763/aber.2009.0408.
- [49] G. Bloom, B. Alsulami, E. Nwafor, y I. C. Bertolotti, “Design patterns for the industrial Internet of Things”, *IEEE Int. Work. Fact. Commun. Syst. - Proceedings, WFCS*, vol. 2018-June, pp. 1–10, jul. 2018, doi: 10.1109/WFCS.2018.8402353.
- [50] J. Queiroz, P. Leitão, J. Barbosa, y E. Oliveira, “Distributing Intelligence among Cloud, Fog and Edge in Industrial Cyber-physical Systems”, en *ICINCO*, 2019.
- [51] “(PDF) Internet of Things (IoT): Definitions, Challenges, and Recent Research Directions”.
https://www.researchgate.net/publication/320532203_Internet_of_Things_IoT_Definitions_Challenges_and_Recent_Research_Directions (consultado el 27 de noviembre de 2022).
- [52] P. Chapman *et al.*, “CRISP-DM 1.0: Step-by-step data mining guide”, 2000.
- [53] L. J. Rojas Bolaños, “Modelo ontológico de usuario para la web semántica de las cosas, basado en técnicas de programación neurolingüística para ayudar a prevenir el síndrome de Burnout”, Universidad del Cauca, 2020.
- [54] M. MENDOZA-MORENO, C. GONZÁLEZ-SERRANO, y F. J. PINO, “Focus Group como proceso en ingeniería de software: Una experiencia desde la práctica”, *DYNA*, vol. 80, núm. 181, pp. 51–60, 2013, [En línea]. Disponible en: <https://revistas.unal.edu.co/index.php/dyna/article/view/36969>
- [55] Eran Feit, “TensorFlowProjects/Brain-Tumor at master · feitgemel/TensorFlowProjects”.
<https://github.com/feitgemel/TensorFlowProjects/tree/master/Brain-Tumor> (consultado el 27 de septiembre de 2022).
- [56] J. Zhao y H. Zhang, “Thin-Plate Spline Motion Model for Image Animation”, pp. 3647–3656, mar. 2022, doi: 10.48550/arxiv.2203.14367.