

API-First, un mapeo de literatura gris para elaborar un marco descriptivo

API-First, a mapping of grey literature to develop a descriptive framework

Sandra Casas ⁽¹⁾, Héctor Reinaga ⁽²⁾

⁽¹⁾Instituto de Tecnología Aplicada - Universidad Nacional de la Patagonia Austral - Unidad Académica Río Gallegos - Santa Cruz - Argentina

scasas@uarg.unpa.edu.ar

Orcid ID: 0000-0002-8289-6132

⁽²⁾Instituto de Tecnología Aplicada - Universidad Nacional de la Patagonia Austral - Unidad Académica Río Gallegos - Santa Cruz - Argentina

hreinaga@uarg.unpa.edu.ar

Orcid ID: 0009-0005-2365-1169

Resumen:

La cantidad de desarrolladores, líderes de IT y empresas que utilizan API-First, crece rápidamente de acuerdo a una encuesta de la empresa Postman. API-First ha surgido en la industria, y desde allí se promueve su adopción, sin embargo, en esta etapa inicial de la metodología; la investigación, documentación y herramientas que ayuden a adoptar un enfoque de diseño API-First son mínimos. El objetivo del trabajo es describir diversos aspectos y características principales del enfoque API-First desde la visión de la industria API y comunidades de desarrollo. Se propone conceptualizar, describir y caracterizar el enfoque desde los aportes de la práctica y sus promotores. No existen métodos específicos para revisar literatura gris (LG), por lo que se aplicaron enfoques conocidos de mapeos sistemático de revisión de la literatura científica y guías para incluir LG. Como resultado, se obtuvieron 42 ítems de LG de fuentes provenientes de la industria y comunidades de desarrollo. Se identificaron tres ideas dominantes para conceptualizar a API-First. Para promover el enfoque, se resaltan principalmente casos de éxito en empresas importantes que aplican API-First. En este artículo, se presenta una lista de nuevas oportunidades y necesidades de investigación del enfoque API-First.

Abstract:

The number of developers, IT leaders, and companies using API-First is growing rapidly, according to a survey by Postman. API-First has emerged in the industry, and its adoption is being promoted, but at this early stage of the methodology, there is minimal research, documentation, and tools to help adopt an API-First design approach. The objective of this work is to describe various aspects and main characteristics of the API-First approach from the perspective of the API industry and development communities. It aims to conceptualize, describe, and characterize the approach based on contributions from practitioners and its promoters. There are no specific methods for reviewing gray literature (GL), so known approaches to systematic mapping of scientific literature reviews and guidelines for including GL were applied. As a result, 42 GL items were obtained from sources in the industry and development communities. Three dominant ideas were identified to conceptualize API-First. To promote the approach, success stories in major companies that apply API-First are highlighted. This article presents a list of new opportunities and research needs for the API-First approach.

Palabras Clave: *APIs web, API-First, mapeo sistemático, industria API.*

Key Words: *Web APIs, API-First, systematic mapping, API industry.*

I. INTRODUCCIÓN

Los productos software de la actualidad, se basan cada vez más en el uso de APIs (application programming interface) web. Las tecnologías de mayor innovación como IA, IoT, Cloud computing, big data, microservicios, apps móviles, blockchain, etc., se consumen o acceden a partir de APIs. Desde la creación de la arquitectura REST [1], la interoperabilidad entre aplicaciones de todo tipo, es una característica central para dar soluciones creativas y eficaces a los problemas de la sociedad.

El aumento exponencial de consumo y oferta de APIs web ha creado una industria API [2], que se diversifica en productos, servicios y recursos, generando beneficios económicos para las empresas, conocido como la Economía API [3] [4]. Las APIs web se han convertido para las empresas en un negocio rentable.

Las APIs web han fomentado innovaciones importantes, entre estas, se observa con cierta distancia, lo que se conoce como “API-First”. API-First, es un concepto aún ambiguo, se presenta como enfoque, estrategia, modelo, proceso, en el que el primer paso consiste en desarrollar las APIs web y luego sobre la base de estas, se construyen los sistemas o aplicaciones software [5]. Todas las capacidades de una organización y sus sistemas, se exponen a través de una API web, estableciendo que la base del diseño del sistema es la definición de APIs claras y bien definidas [6].

API-First es una metodología emergente, que está atrayendo la atención de los profesionales del software en la industria [6]. La cantidad de desarrolladores, líderes de IT y empresas que utilizan API-First crece año a año, de acuerdo a las últimas encuestas de la industria API de Postman [7] [8]. API-First ha surgido en la industria, y desde allí se promueve su adopción, según [6] API-First está en la infancia, por eso en esta etapa inicial de la metodología, la investigación, documentación y herramientas que ayuden a adoptar un enfoque de diseño API-First son mínimos.

En este trabajo, se presenta un mapeo sistemático de literatura gris (LG), cuyo objetivo es describir y caracterizar a API-First. Desde fuentes provenientes de la industria y comunidades de desarrollo, se recupera, analiza y clasifican algunas características de API-First, con el propósito de colaborar en la construcción de un encuadre conceptual más académico y formal. Para el desarrollo de este estudio se aplicaron directrices metodológicas de [9] [10].

Se obtuvieron 42 ítems de LG. Se identificaron tres ideas dominantes para conceptualizar a API-First, que se presenta con diversas variantes, donde Design-First y Code-First son las más populares y se mencionan en contraposición. Ocho beneficios y algunas de sus causas se destacan, donde la “mayor consistencia en el desarrollo” tiene un impacto causal en las restantes ventajas. Existen prácticas, actividades y características que relacionan claramente a API-First con métodos ágiles y DevOps como la integración continua. Para promover el enfoque se resaltan principalmente casos de éxito, sucedidos en empresas importantes que aplican API-First. El 57% de los ítems proporcionan guías generales o específicas para adoptar API-First, pero éstas son demasiado heterogéneas. API-First requiere herramientas para dar soporte al ciclo de vida API, donde OpenAPI y Postman son las más recomendadas, pero también requiere herramienta de soporte a la administración y gobernanza de APIs. Los impulsores también señalan al menos tres requisitos para adoptar este enfoque, relacionados a cuestiones culturales, organizacionales y técnicas.

Este estudio inicia la descripción del enfoque API-First, sin embargo, es necesario profundizar diversos aspectos que aún son desconocidos. En este sentido, se plantea una agenda de ideas que pueden motivar nuevas líneas de indagación para continuar el estudio y comprensión de API-First.

Este artículo tiene la siguiente estructura: en la Sección 2 se describen las motivaciones que impulsan este estudio, en la Sección 3 se desarrolla la metodología aplicada, en la Sección 4 se presentan los resultados, y en la Sección 5 se plantean las discusiones y se obtienen las conclusiones.

II. MOTIVACIONES

Los tópicos más abordados en la investigación sobre APIs web hacen foco en el dominio tecnológico, particularmente usabilidad, evolución, documentación, gestión, especificación y diseño. [11] [12] [13] [14] [15] [16]. En estas revisiones no se menciona API-First.

El enfoque API-First comenzó a ganar popularidad alrededor del año 2016 [17]. Existen algunos registros anteriores, como los blogs, “The Emergence of API-First Development de Robles” en 2014 y “How to Design Great APIs with API-First Design de Riggins” en 2015, publicados en el desaparecido sitio programmableweb, según citas de [18]. Estos indicios revelan que API-First

tiene al menos 10 años, pero aún son insuficientes estudios descriptivos.

En [6] se identifica que el desafío importante asociado con API-First radica en la etapa inicial de conocimiento del tema en que se encuentra y en consecuencia en la necesidad de una investigación revisada por pares que defina pautas para su adopción y una base para futuras investigaciones. Los autores buscan hacer el estado de la metodología de diseño API-First mediante la exploración de publicaciones de la comunidad académica y la LG disponible. Consideramos muy inicial este trabajo y los resultados, dado que se basa en 3 artículos científicos y 10 ítems de LG, provenientes de diversos blogs de la web.

Existen algunas tesis y tesinas que abordan el tema. En [18] el objetivo principal es comprender qué es el API-First Design, sus beneficios y cómo aplicar este método en el entorno de Azure API Management, utilizando herramientas de terceros para hacer que el proceso sea eficiente. Y en [19] se describe cómo implementar prácticas de gobierno de APIs en una empresa de servicios públicos. Utilizan la gestión de APIs para la interacción entre aplicaciones (internas y externas) empleando el enfoque API-First con el fin de reducir los tiempos, la complejidad, y los costos en el desarrollo de APIs en los equipos de trabajo de la empresa. Luego en [20] estudian el valor del enfoque "API-First" al desarrollar APIs, comparándolo con el enfoque "Code-First".

El informe de Postman del año 2023 [7], basado en una encuesta a 40.261 desarrolladores y profesionales de API de todo el mundo y datos agregados y anónimos de la plataforma API de Postman, confirma que, (a) la cantidad de desarrolladores, líderes y empresas que utilizan API-First crece rápidamente, (b) los adoptantes de API-First producen APIs más rápido y encuentran menos fallas que el promedio. El informe de Postman del año 2024 [8] basado en una encuesta a 5.600 desarrolladores y profesionales de API de todo el mundo, ratifica las tendencias mencionadas, ya que indica que, el 74% de los encuestados en 2024 usan API-First, contra el 66% del año 2023, y que los beneficios de API-First son tangibles: (a) Producción de APIs más rápida: los desarrolladores podrían producir una API en una semana (63 % de los encuestados), y (b) Recuperación rápida de fallas: las organizaciones API-First se recuperan de las fallas de API más rápido, a menudo en una hora.

Los escasos antecedentes en relación a API-First, más allá de la recopilación inicial de [6], fundamentan la necesidad de realizar una indagación más profunda y exhaustiva,

para proponer una descripción más completa y vislumbrar nuevas líneas de trabajo.

III. METODOLOGÍA

Este estudio se estructuró en cuatro etapas principales. En la primera etapa se definieron los objetivos y las preguntas de investigación. La segunda etapa consistió en la búsqueda y selección de ítems, para lo cual se definieron primero los protocolos y luego se pasó a la ejecución de los mismos. La tercera etapa, consistió en la extracción de datos, que se hizo en varias pasadas. En el cuarto y último paso, se realizó el análisis y síntesis de la información obtenida para responder las preguntas formuladas. Dado que este trabajo solo abarca literatura gris (LG), para la cual no existen métodos específicos de revisión, metodológicamente se usaron las guías de [9] [10], siempre que fue posible.

La decisión de analizar ítems (artículos o blogs o post) de la red, proveniente de empresas líderes en IT y expertas en APIs, revistas profesionales y sitios especializados, responde a varios factores que consideramos claves, (a) API-First está más desarrollada en la industria y desde allí se promueve con fuerza, (b) los autores y las audiencias de estos ítems, son expertos, profesionales, desarrolladores, gerentes, que siguen o están interesados en este tópico con intención técnica o de negocios y conforman la comunidad de desarrollo y el sector de la industria, (c) el contenido de los ítems es técnico.

Este estudio plantea un desafío ya que los ítems son textos que se caracterizan por ser cortos, con pocas secciones o apartados y no responden a una estructura en común como IMRD. Como géneros literarios corresponden a ensayos, y según el contenido e intencionalidad del autor serán argumentativos, explicativos o descriptivos o una combinación de estos. Estos textos no fueron escritos para ser analizados científicamente. Por estas características, responder a algunas preguntas requirió leer todo el texto y componer desde distintos fragmentos las respuestas.

Objetivos y Preguntas de Investigación

El objetivo de este estudio es describir diversos aspectos y características principales del enfoque API-First desde la visión de la industria API y comunidades de desarrollo. Para ello nos proponemos conceptualizar, describir y

caracterizar el enfoque desde los aportes de la práctica y sus promotores. En este sentido, y de acuerdo a la categorización de preguntas de [21] y [22] se formularon consignas de orden exploratorio y descriptivo. Se plantearon tres preguntas generales, y a cada una de ellas se las descompuso en preguntas más específicas.

P#1: ¿Qué es API-First?

P#1.1: ¿Cómo se define API-First?

P#1.2: ¿Cuáles son las variantes o tipos de API-First?

P#2: ¿Por qué aplicar o adoptar API-First?

P#2.1: ¿Cuáles son las ventajas o beneficios de API-First?

P#2.2: ¿Se relaciona API-First con otros enfoques de desarrollo como métodos ágiles o devops?

P#2.3: ¿Se relaciona API-First con las actividades, la administración de API o gobierno de API?

P#2.4: ¿Se presentan, describen o mencionan casos de éxito o casos de uso o casos de estudio o escenarios o ejemplos reales o experiencias, de la aplicación o adopción de API-First?

P#3: ¿En qué consiste aplicar API-First?

P#3.1: ¿Se brinda una guía o pasos o lineamientos para aplicar API-First?

P#3.2: ¿Qué herramientas requiere API-First?

P#3.3: ¿Cuáles son los requisitos o precondiciones para adoptar API-First?

La pregunta P#1 pretende llegar a la elaboración de un concepto, y distinguir las distintas formas que existen de API-First, y surge principalmente de la comparación en [20]. La pregunta P#2 busca comprender por qué API-First es promovida e impulsada desde la industria, relevar evidencias prácticas y/o empíricas de la aplicación o adopción de API-First, y qué antecedentes tiene, a partir de la relación con otras actividades de desarrollo de software en general (métodos ágiles y DevOps) y de API en particular (administración y gobierno API). El interrogante P#3, amplía la pregunta P#1, puesto que propone llevar a un nivel más concreto que es API-First, al describir sus componentes principales.

Búsqueda y Selección

El protocolo de búsqueda y selección consistió en la definición de la cadena de búsqueda, la fuente o base de

datos en la que se aplicó, y los criterios de inclusión y exclusión para identificar los ítems relevantes.

Dado los objetivos y preguntas de investigación formulados, se decidió crear la siguiente cadena de búsqueda: ("API First" OR "API- First"). La cual es amplia y no restrictiva, dado que el uso de cadenas posibles como "API-First design", "API-First development" o "API-First Strategy" (entre otras), que son otras formas de denominar el concepto, estarían incluidas.

La relevancia se determinó utilizando los siguientes criterios de inclusión:

(I1) el ítem es un blog, white papers, informe gubernamental, artículo revistas

(I2) el ítem se refiere a API-First.

(I3) el ítem está escrito en inglés

(I4) el ítem cumple con los criterios de relevancia y credibilidad definidos.

Además, se excluyeron los ítems de LG que cumplieran los siguientes criterios:

(E1) el ítem del GL no refiere API-First.

(E2) el texto no está escrito en inglés.

(E3) el ítem corresponde a un post de una red social (Linkedin, X, Instagram, etc.) o repositorio (GitHub, StackOverflow, etc.).

(E4) el formato del ítem no es texto (videos, podcast, etc.).

Dado que este tipo de elementos de LG no sigue el proceso de revisión controlado como la publicación científica, la calidad puede variar e incluso es difícil de determinar. Por otro lado, no es requisito aplicar evaluación de calidad a los mapeos. Por ello, se adoptó una postura para la inclusión de ítems y es la que se denominó relevancia y credibilidad (I4). Para determinar estas propiedades se usaron algunas dimensiones y condiciones propuestas por [9] para evaluar la calidad de LG. A continuación, se detallan:

El ítem es aceptable, sí: (1) el sitio o portal web donde el ítem está publicado corresponde a una empresa líder de APIs o IT, o (2) corresponde a una Revista de IT para profesionales/desarrolladores, o (3) es un Sitio de Blogs especializados en APIs. En los restantes casos, el artículo debe explicitar: fecha, autor, bio, e impacto. El impacto lo medimos con el pagerank de la página (<https://dnschecker.org/pagerank.php>), y sólo se admiten

los que midieron seis o más puntos, en un rango de cero a diez.

Búsqueda

Se aplicó la cadena de búsqueda al motor de búsqueda de Google en el mes de julio de 2024, usando la opción de búsqueda avanzada se aplicaron dos filtros: realizar la búsqueda de la cadena en los títulos de las páginas y recuperar sólo ítems en idioma inglés. Esta búsqueda arrojó un resultado de 6390 ítems.

Se extrajeron los resultados de las primeras 20 páginas (10 ítems por página). Luego de eliminar los duplicados (39), y aplicar los criterios de inclusión y exclusión (119), resultaron 42 ítems a examinar. Se decidió detener la búsqueda en las 20 primeras páginas dado que a posterior se comenzaron a recuperan ítems repetidos y de bajo pagerank.

Extracción de Datos

Se elaboró una planilla con las preguntas en las columnas y las filas correspondían a cada ítem. Se procedió a la lectura de los 42 ítems obtenidos del paso anterior y se fue respondiendo por pregunta, registrando la respuesta en la planilla. La división entre los autores fue por pregunta, (la autora 1 contestó las preguntas 1 2 y 3 y las restantes el autor 2). Luego, cada autor revisó las respuestas del otro autor. Como ya se indicó, dado que los ítems no tienen estructura, para responder una consigna, se requiere leer todo el texto. Esto llevo a varias revisiones.

Análisis y Síntesis

El conjunto de respuestas registrado en la planilla, se analizó de diversas formas. Análisis de mayor frecuencia de aparición del mismo término o concepto, comparación con otras categorizaciones o clasificaciones o definiciones, enumeración de datos, fueron las más usadas.

IV. RESULTADOS

En esta Sección se presentan los hallazgos de la ejecución del mapeo sistemático, los cuales corresponden a la selección de ítems enunciados en el apéndice de referencias de estudios (Anexo).

La Tabla 1 presenta la distribución de los 42 ítems por tipo de fuente, y la Tabla 2 por año.

Tabla 1. Distribución de ítems según la fuente.

Origen del Ítem	Cantidad
Empresa líder de APIs o IT	13
Revista IT de profesionales/desarrolladores	5
Sitio de Blogs especializados en APIs	5
Otro	19
Total	42

Tabla 2. Distribución de ítems por año.

Año	2015	2017	2018	2019	2020	2021	2022	2023	2024
Cant.	2	2	1	2	3	5	11	11	5

En el Anexo se detalla la ubicación o dirección de estos ítems. A continuación, se responden los interrogantes planteados anteriormente.

P#1: ¿Qué es API-First?

En primer lugar, ante la pregunta ¿cómo define API-First?, se han analizado las ideas principales que comunican las definiciones propuestas. La Tabla 3, presenta las tres ideas más expresadas en las definiciones.

Tabla 3. Definición de API-First.

Ideas Dominantes	Ítem
Enfoque de desarrollo de software.	I2 - I9, I11-I16, I19, I21-I26, I27-I42
Las APIs se desarrollan al inicio, antes que las aplicaciones o sistemas.	I1, I3-I19, I21-I30, I32-I42
Las APIs son el componente o producto central y principal (del negocio, la arquitectura y sistema).	I1-I2, I4, I6-I25, I27-I29, I31, I32, I34, I37, I40, I41

Las definiciones provistas también explicitan que en este enfoque de desarrollo, el diseño y la especificación de la API se priorizan, es decir son el primer paso [15, 17, 118, 119, 122, 128, 130, 133, 136, 137], obteniendo la documentación o contrato que se usará para generar o guiar el desarrollo de código [15, 18, 126, 128, 130]. Otras formas similares de conceptualizar son, modelo de desarrollo [132, 135] proceso de desarrollo [118, 140] y metodología de desarrollo [117].

Varios ítems asocian la definición de API-First con las arquitecturas de software desacopladas y distribuidas

implementadas en nubes privadas o públicas, específica y principalmente microservicios, [13, 15, 17, 114, 116, 121, 122, 123, 126, 131, 132, 133, 134, 138, 142].

En relación a la pregunta ¿cuáles son los enfoques o tipos de API-First?, en todos los casos se considera que API-First es una estrategia en la cual la API se desarrolla primero, existen algunos ítems que presentan variantes del enfoque. En [11, 13, 115, 122, 132, 135, 138, 139] las variantes API Design-First y API Code-First se presentan como enfoques en contraposición. API Design-First se centra en definir y diseñar la API utilizando herramientas como OpenAPI o RAML, permitiendo la creación de mock servers y la iteración del diseño sin necesidad de escribir código desde el principio. Es un enfoque más colaborativo y permite una mayor iteración y consistencia en el diseño de APIs. API Code-First prioriza el desarrollo del código de la API desde el principio. Es más común entre desarrolladores con un enfoque en ingeniería, ya que permite un despliegue más rápido una vez que la API ha sido iterada y completada.

Sólo [139] expone la existencia de las alternativas, API Prototype-First y API Proxy-First. Prototype-First consiste en que se desarrollan y documentan prototipos de la API usando herramientas antes de generar una definición formal de la API. Proxy-First propone que se genere una API a partir del tráfico existente a través de proxies o interceptores.

En [16, 137] se analiza la variante API Integration-First, como un enfoque demasiado lento, que se centra únicamente el proyecto específico y crea una API, sin pensar en su utilización futura. Es un enfoque tradicional ascendente en el que primero se construye la lógica de integración sobre los servicios backend y luego se exponen los servicios integrados, como API administradas a los usuarios.

Los términos API-First como API Design-First se usan en forma indistinta para denotar el mismo enfoque [13, 121], enfatizan la importancia de los documentos y contratos de diseño. Los términos API-First y API Design-First suelen usarse indistintamente, pero no son exactamente lo mismo [3].

En [119] se indica que API-First y API-as-a-Product, son términos análogos, siendo un error. Como ya se indicó, una empresa API-First, prioriza las APIs, incluso si las APIs que construye son estrictamente para fines internos. API-as-a-Product, es cuando toda la empresa se construye en torno a las APIs. Las APIs constituyen el modelo de negocio principal y, por lo general, se ofrecen como un servicio público dirigido a los desarrolladores como el principal consumidor. Es decir, todas las empresas API-as-a-Product son API-First, pero no todas las empresas API-First son API-as-a-Product.

Por último, [17] distingue que “API-First” es un desarrollo de arriba hacia abajo, y “ESR-First” es un desarrollo de abajo hacia arriba. El enfoque “API-First” implica diseñar la API antes que cualquier otra funcionalidad, mientras que el enfoque “ESR-First” implica construir las APIs basadas en servicios ya existentes.

P#2: ¿Por qué aplicar o adoptar API-First?

Este interrogante se descompuso en cuatro preguntas. En la Tabla 4, se indican los ocho beneficios de adoptar API-First. Se han extraído los beneficios más frecuentemente mencionados y qué práctica o actividad o característica, causa o se relaciona con el mismo. Se observa que la mayoría de estos beneficios son dependientes y se relacionan, al igual que las causas asociadas.

Tabla 4. Beneficios de API-First.

Beneficios	Causa	Ítem
<i>Eficiencia y productividad</i> Iteraciones más rápidas – Aceleración del desarrollo y despliegue de aplicaciones – Lanzamiento y entrega más rápido de productos o funcionalidades	• Desarrollo en paralelo de los equipos. • Usar herramientas para generar código, SDKs, y pruebas automáticamente desde las especificaciones. • Reducción de riesgos y deuda técnica. • Reutilización de APIs en diferentes proyectos. • Flexibilidad.	I1, I2, I3, I6, I8, I9, I12, I13, I15, I17-I19, I20, I22-I27, I30, I31, I34, I37, I39, I42
<i>Mayor consistencia en el desarrollo</i>	• La especificación temprana de las APIs se convierte en un contrato estandarizado para todo el desarrollo y equipos participantes.	I1, I3, I4, I8-I10, I15, I17-I22, I25, I28, I30, I33, I35
<i>Mejor colaboración y comunicación entre diferentes equipos</i>	• Se proporciona una base clara y acordada desde el inicio del proyecto que permite la alineación entre equipos. • Al establecer las APIs como contratos, todos los equipos involucrados tienen una comprensión clara de cómo deben interactuar los diferentes componentes.	I1, I3- I5, I7, I9, I10, I12, I14, I15, I17-I19, I21-I24, I26, I28, I32, I33, I35-I37, I39-I42
<i>Mejora de la calidad de APIs, del software, del producto final.</i>	• APIs bien diseñadas y bien documentadas. • La especificación detallada y el diseño anticipado resultan en APIs de mayor calidad. • La estandarización y reutilización de APIs. • La gestión del cambio de las APIs y mantenimiento es más fácil, asegurando que la documentación esté siempre sincronizada con la implementación.	I1, I3, I7, I13, I19, I21, I33, I36, I39, I42
<i>Menor costo de desarrollo</i> Reducción de costos	• Detección y corrección de errores en fases tempranas. • La reutilización de APIs y la integración sencilla. • Flexibilidad. • Pruebas.	I2-I4, I12, I27, I29, I34, I35
<i>Escalabilidad</i> fácil adaptación de nuevos servicios y la evolución de la arquitectura del sistema	• Al tener un núcleo bien definido y estandarizado para la comunicación entre equipos. • Reutilización: Permite que las APIs se reutilicen en diferentes contextos. • Consistencia. • Flexibilidad.	I4, I8, I11, I14-I16, I18, I20, I25-I27, I32-I35, I38, I41
<i>Mejor experiencia del desarrollador/ cliente</i>	• Las APIs claras, bien documentadas y diseñadas, resultan más fáciles de usar y entender, mejoran la usabilidad y reducen la complejidad para los desarrolladores.	I2, I6, I8, I11, I14, I25, I32, I37, I40
<i>Mejora la integración e interoperabilidad</i> entre diferentes sistemas, servicios y aplicaciones, especialmente en infraestructuras distribuidas.	• Consistencia en el diseño, documentación clara, y buen diseño de la API. • Automatización de pruebas	I7, I10, I11, I16, I18, I19, I21, I23, I25, I26, I28, I29, I32, I34, I37, I41

En la Tabla 5, se responde la pregunta, ¿se relaciona API-First con otros enfoques de desarrollo como métodos ágiles o DevOps?. Para elaborar esta respuesta los textos se han analizado de varias formas, dado que explícitamente no se enuncia una relación. Se han identificado características o prácticas en común, que se

mencionan entre beneficios o ventajas, pasos y actividades en guías.

Tabla 5. Prácticas comunes entre API-First y Ágiles - DevOps.

Característica/practica	Ágil	DevOps	Ítem
Iteración rápida	✓		I1, I4, I8-I10, I14, I15, I18, I19, I21-I25, I27-I30, I32
Entrega continua	✓	✓	I8, I10-I12, I14, I15, I18, I22, I23, I25, I26, I29, I31-I35, I37, I38
Integración Continua	✓	✓	I7-I10, I12, I13, I17-I19, I21-I23, I25-I28, I30, I32-I34, I42
Despliegue Continuo		✓	I1, I3, I4, I12, I17, I19, I21, I22, I30, I31, I33
Pruebas automáticas tempranas	✓		I13, I20, I21
Comunicación y colaboración de equipos	✓	✓	I1, I3, I4, I7, I12, I17, I23, I26, I29, I34, I39, I42
Trabajo Paralelo		✓	I11, I36
Automatización		✓	I9, I12, I16, I29, I31
Planificación	✓		I18, I19

En [19] se resalta que API-First mejora la agilidad, en [115] se indica que las metodologías ágiles son útiles con API-First. En [128] analiza que API-First hace más ágil al proceso ágil y compara los procesos aportando gráficos. La razón más importante para usar API-First es el aumento de la agilidad y la reducción del tiempo de desarrollo desperdiciado al crear APIs. Concluyen que con API-First, se posibilita redactar borradores, cambiar iterativamente y proporcionar comentarios sobre el diseño más rápido.

En [110] se hacen referencias a características de DevOps, ya que explica que el proceso API-First requiere la participación de diferentes equipos y herramientas y que es adoptado por equipos que tienen una amplia experiencia en la creación de flujos de trabajo de CI/CD que pueden automatizar la validación de la APIs. En [131] en la guía para adoptar API-First,

uno de los pasos indica, “Crear una cultura API-First”, la que consiste en cambiar la comunicación y enfoque en los equipos de DevOps, DevSecOps, producto e ingeniería para permitir que las APIs tomen la iniciativa. En [120] sostiene que API-First ha surgido de los contextos de ágiles y DevOps, estableciendo una relación y consecuencia.

En cuanto a la pregunta ¿se relaciona API-First con las actividades, la administración de API o gobierno de API?, se analizó la definición de Gobernanza de API de [23], para luego establecer las relaciones con los datos aportados por los ítems.

Según [23], la gobernanza de API proporciona un enfoque basado en políticas que ayudan a aplicar estándares y puntos de control durante todo el ciclo de vida de la API. Abarca no sólo el tiempo de ejecución de la API, sino también el diseño y los procesos de desarrollo. Incluye las pautas, estándares y procesos que se deben seguir para la identificación de API, la documentación de la interfaz, el desarrollo, las pruebas, la implementación, la ejecución y la operación. Los estándares y principios definidos por la gobernanza de API, brindan garantía de calidad de API, como seguridad, disponibilidad, escalabilidad y confiabilidad.

API-First al establecer el diseño y documentación de la API, como el primer paso, a partir de una especificación como OpenAPI, establece no sólo un contrato, sino un estándar. Cumpliendo un principio de la gobernanza clave para el ciclo de vida de la API. Luego explícitamente, existen referencias más marcadas. En [123] presenta un proceso de tres fases para crear una cultura y una empresa API-First. Hay varios pasos, desde el diseño y las reglas de la estructura y el desarrollo de las API, hasta la priorización y la planificación del tipo correcto de infraestructura para respaldar las API. En la primera fase, incluye pasos que consisten en la administración de APIs.

En [124] indica que la gobernanza es importante en los contextos de diversidad tecnológica y seguridad, ya que permite supervisar y orientar los desafíos a los que se enfrenta la plataforma API y hacer cumplir los estándares en toda la plataforma API. Sin una gobernanza adecuada, se debilita todo el concepto de API-First.

En [130] señala que API-First necesita herramientas de gestión de API para: aplicar protecciones de seguridad,

o las capacidades más avanzadas pueden requerir una plataforma de gestión de API de ciclo de vida completo, mecanismos para hacer que las API estén disponibles como productos de API a través de portales de autoservicio, opciones de monetización, etc.

En [133] infieren que los modelos API-First también simplifican la gobernanza de API, brindando más control y observabilidad a los equipos de operaciones de forma predeterminada. Tener un mayor control y visibilidad en la API, permite a los equipos visibilizar el estado actual de la API como su potencial futuro.

La última consigna de esta pregunta, en relación a si se presentan, describen o mencionan, casos de éxito o casos de uso o escenarios o ejemplos reales o experiencias, de la aplicación o adopción de API-First, se contesta en la Tabla 6. Los casos de éxito relacionan el éxito de una empresa con el hecho de aplicar API-First, mientras que los ejemplos proponen el desarrollo de la especificación OpenAPI, uso de herramientas, generación código de la API, pruebas de problemas.

Tabla 6. Casos de éxito, prueba de concepto, y caso problemas.

Tipo	Descripción
Caso de éxito	Twilio, Stripe, Segment, Netflix, Amazon, PayPal, Plaid, SCR, Cargill, Digital Twins, Cisco, Schweiz o Scout24, Commercetools.
Ejemplo	Definiciones de OpenAPI (OAS) con un módulo de servidor Express. Diseño de una API de chistes (Joke API) utilizando SwaggerHub. Utilización de API basadas en OAS para comunicarse entre microservicios. Simulaciones de esquemas y comportamientos de API. Proyecto de API bancaria. Aplicación móvil y web.
Prueba de Concepto	Demostración de la detección de problemas con la especificación API desde el diseño inicial con OpenAPI (validación de la especificación).
Caso problema	Se identifica el problema de hacer los datos y la funcionalidad de la operación de un negocio existente, accesible a cualquier aplicación. Propone utilizar una API para acceder al contenido y los datos, invirtiendo en las siguientes áreas: aplicaciones heredadas, nuevas aplicaciones, y portal para desarrolladores.

Una nota aparte surge de [126] y [138], los cuales referencian a la Alianza MACH, la cual promueve el desarrollo de aplicaciones modernas a través de ecosistemas de tecnología abierta. MACH es un acrónimo que representa Microservicios, API-First,

Cloud-native (SaaS) y Headless. Los miembros de esta alianza son un conjunto importante de empresas de la industria del software, muy reconocidas, que buscan ofrecer soluciones para crear aplicaciones altamente flexibles y escalables.

P#3: ¿En qué consiste aplicar API-First?

Para responder a esta pregunta, las formas de aplicar o adoptar API-First se han clasificado en guías generales y guías específicas. Las guías generales, son directrices o lineamientos que orientan el desarrollo o aplicación del enfoque. En cuanto a las guías específicas, se componen de un conjunto de pasos entre los cuales se establece un orden de ejecución, incluso pueden estar divididos en etapas y/o fases.

Siguiendo esta clasificación en [11, 13, 15, 19, 110, 112, 115, 122, 123, 124, 128, 129, 131, 137, 139] se proporcionan guías específicas para adoptar API-First, usando herramientas OpenAPI y Postman. Y en [12, 14, 16, 17, 114, 126, 130, 140 142] se proporcionan guías más generales. En las Tablas 7 y 8, se transcriben algunas de las guías aportadas, en las cuales se observa que las tareas o pasos son de características muy heterogéneas.

Tabla 7. Guías generales

I40	I42
Al inicio, las partes técnicas y no técnicas deben poder discutir las funcionalidades de la API, definir la estructura del contrato, realizar simulaciones operativas y concebir las pruebas de validación. Luego, se implementará el contrato de API aprobado y, los desarrolladores, así como los evaluadores de calidad, podrán implementar todas las aplicaciones y pruebas en paralelo, con soporte de documentación adicional y entornos de prueba.	Los equipos de desarrollo deben documentar las funciones y los datos necesarios para lograr una determinada tarea, resolver un problema o agregar una capacidad particular al software existente. Esto servirá como una lista de las funciones y los servicios clave que las APIs deberán exponer. Finalmente, los equipos pueden comenzar a escribir los bloques de código que conformarán la API y, por último, la aplicación.

Tabla 8. Guías específicas.

I3	1. Definir y documentar las APIs utilizando especificaciones como OpenAPI. 2. Crear y aplicar guías de estilo para mantener la consistencia. 3. Utilizar herramientas como Stoplight Studio para diseñar, documentar y validar APIs.
I23	Fase 1. Visualice y Evalúe Paso 1: Crear principios rectores Paso 2: Visualizar el destinatario Paso 3: Realizar un inventario Paso 4: Determinar infraestructura y requerimientos de colaboración Fase 2. Diseñar y Crear Paso 5: Crear principios de diseño Paso 6: Crear reglamento de API Paso 7. Establecer infraestructura Fase 3. Elevar y educar Paso 8: Elevar los proyectos faro. Paso 9: Crear un embajador y programas de educación.
I31	1. Realizar un inventario: comprender qué API tiene, cuáles le faltan y cómo (y dónde) se utilizan; esto incluye las API públicas, privadas y de socios. 2. Analizar el flujo de trabajo de desarrollo de API: observe los procesos existentes y comprender qué falta o qué mejoras necesita. 3. Examinar el perfil empresarial: observar los grupos de usuarios y la estructura organizativa y trazar los límites de su organización. 4. Crear una cultura que priorice las API: cambiar la comunicación y enfoque en los equipos de DevOps, DevSecOps, productos e ingeniería para permitir que las API tomen la iniciativa. 5. Seleccionar una plataforma de desarrollo y seguridad de API sólida que ayude a estandarizar y ejecutar.
I39	1. Inventariar y entender las APIs existentes. 2. Definir los límites del dominio de negocio y mapear la estructura organizacional. 3. Adoptar una plataforma de APIs y estandarizar su uso. 4. Entrenar a los equipos en prácticas API-First. 5. Crear y documentar la primera API siguiendo estos principios.

En [I29], [I38] y [I40] se grafica el mismo esquema abstracto y genérico (Figura 1), que ilustra el proceso que se puede realizar en paralelo por diversos equipos

una vez que se define la especificación de la API, marcado por las pruebas. A partir de la especificación, el esquema inicia con el diseño de los mocks y test en un entorno de desarrollo API, continúa con la codificación de la API y finaliza con la integración de la API en un repositorio de código, para dar lugar al despliegue de la API.

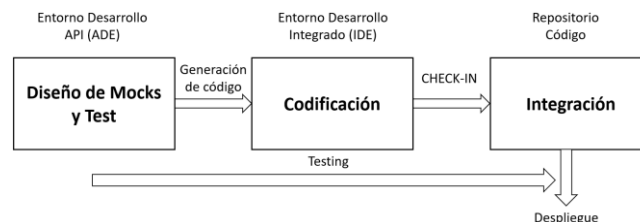


Figura 1. Pasos para una aplicación API-First [I29, I38, I40].

En cuanto al interrogante ¿qué herramientas software requiere API-First? La Tabla 9 presenta las herramientas más usadas y su función.

Tabla 9. Herramientas de soporte.

Función	Herramientas	Ítem
Diseñar, documentar, y probar	• OpenAPI: Para definir y documentar APIs. • Swagger: Para generar y visualizar la documentación de la API. • Postman: Para pruebas y validación de APIs.	I1, I5, I7, I9, I10, I12, I15, I22, I28-I30, I39
Especificar y diseñar	OpenAPI - Stoplight Studio	I3
	RAML	I39
Generadores de SDK	OpenAPI Generator	I4
Simular y probar las APIs	Mock Servers	I1, I21, I39
	SoapUI	I39
Administración y Gobernanza de APIs	Kong Konnect –Treble AWS Gateway	I4, I16

En [I30] señala que API-First necesita herramientas de gestión de API para: aplicar protecciones de seguridad, o las capacidades más avanzadas pueden requerir una plataforma de gestión de API de ciclo de vida completo, mecanismos para hacer que las API estén

disponibles como productos de API a través de portales de autoservicio, opciones de monetización, etc.

El interrogante sobre ¿qué requisitos o precondiciones son necesarios para adoptar el enfoque API-First? en principio se rescatan con claridad tres: (1) Son necesarios el compromiso y/o una cultura organizacional que valore a las APIs como un activo central [11, 13, 17, 112, 122, 123], (2) La capacitación de los equipos y conocimiento en el uso de las herramientas adecuadas para soporte de la definición de estándares claros y el establecimiento de procesos para el diseño y la revisión de APIs [13, 17, 122, 130], y por último (3), las condiciones de trabajo colaborativo de los equipos [11, 13, 17] e integración continua, que redundan en mejor comunicación y trabajo en paralelo.

El enfoque API-First es ideal para proyectos nuevos, donde es crucial un buen diseño de la API desde el principio. Sin embargo, puede no ser adecuado para equipos pequeños o proyectos que requieran mucha experimentación. También puede ser difícil de implementar en organizaciones con una cultura de desarrollo deficiente o con una gran cantidad de burocracia [12].

V. DISCUSIÓN Y CONCLUSIONES

Se obtiene una alta coincidencia (más del 79%) entre los ítems, en que API-First es un enfoque de desarrollo de software, en el cual las APIs se desarrollan al inicio, antes que las aplicaciones o sistemas, y las APIs son el componente o producto central o principal (del negocio, la arquitectura y/o el sistema). El 36% de los ítems asocia API-First con el desarrollo de arquitecturas de microservicios.

Se identifican siete variantes de API-First, las más populares son API Design-First y API Code-First.

Los beneficios más citados son: eficiencia y productividad, mayor consistencia en el desarrollo, mejor colaboración y comunicación entre diferentes equipos, mejora de la calidad, menor costo de desarrollo, escalabilidad, mejor experiencia del desarrollador, mejora la integración e interoperabilidad. Estos beneficios están relacionados entre sí, pero la mayor consistencia en el desarrollo impacta positivamente y posibilita en gran medida el logro de los restantes beneficios identificados. De esta forma, la consistencia en la especificación, diseño y documentación de las APIs se vuelve clave en API-First.

API-First se alinea y complementa bien con diversas prácticas y actividades de los métodos ágiles y DevOps, como la integración continua. De manera similar, API-First se alinea e integra a la Gobernanza y Administración de APIs, sin embargo, estas interacciones no son claras o explícitas.

El 45% de los ítems para resaltar la aplicabilidad del enfoque utiliza casos de éxito, ejemplos, pruebas de concepto o casos problema, se destacan principalmente los casos de éxito (26%), sucedidos en empresas importantes que aplican API-First. En el 57% de los ítems, los promotores proporcionan guías generales o específicas para adoptar API-First, pero éstas son demasiado heterogéneas en los pasos o acciones, se mezclan cuestiones técnicas y operativas con cuestiones de índole más estratégico y/o gerencial y/o de negocios.

Las herramientas para dar soporte a API-First más elegidas y utilizadas son OpenAPI y Postman. Las herramientas necesarias en API-First se clasifican en, las que dan soporte al ciclo de vida de API (diseño, pruebas, documentación), pero también son incluidas herramientas para la administración y gobernanza, que dan soporte a funciones de monitoreo, seguridad, etc.

Se han identificado tres precondiciones para aplicar o adoptar API-First, estas son de orden técnico, organizacional y humano-social. Sin embargo, y en consideración a las consignas anteriores, permiten suponer que haber adoptado métodos ágiles y DevOps son condiciones auspiciosas para una adopción de API-First exitosa.

Finalmente, se observa que existe una cierta coherencia o correspondencia entre diversas respuestas. En el caso de las relaciones identificadas con la administración y gobierno de APIs (P#2), luego se condicen con la necesidad de herramientas para dar soporte a las mismas (P#3).

El beneficio “Mejor colaboración y comunicación entre diferentes equipos” (P#2), aparece con las relaciones con métodos Ágiles y DevOps y luego se configura como uno de los requisitos para aplicar API-First (P#3).

Limitaciones

Aunque se han seguido las pautas de [9] [10] para realizar este mapeo, el proceso puede presentar limitaciones. Es posible que se hayan introducido sesgos durante el proceso de selección de los ítems de LG que causaron problemas como: errores en la

extracción de datos, insuficientes detalles en las fuentes para realizar las descripciones y/o clasificaciones completamente. Para evitar sesgos, la selección de los ítems fue realizada por los dos autores en forma separada. Después de la selección, se compararon los resultados para resolver las diferencias.

Además, existen algunas situaciones fuera de nuestro control que pueden interferir en la calidad o precisión de la investigación. Dado que son textos surgidos de blogs cuyo propósito no fue el que se dio en este estudio, se ha observado; la falta de detalles suficientes sobre las afirmaciones o datos que se exponen; no se abordan cuestiones de validez; y los métodos de recopilación y análisis de datos no siempre revelan detalles importantes. Estas características interfirieron en nuestro proceso de análisis, por lo que los investigadores tomamos algunas decisiones subjetivas para realizar la síntesis de los datos. Para minimizar estas limitaciones, los investigadores discutieron todas las dudas para resolver todas las ambigüedades.

Nuevas oportunidades y necesidades de Indagación

Los incipientes resultados hallados exponen nuevas oportunidades de estudio: (1) los promotores de API-First se apoyan en los beneficios del enfoque y en los casos de éxito para recomendar la adopción del enfoque. Aunque estos aspectos son importantes, es necesario identificar y definir con mayor claridad y profundidad las capacidades y prácticas de API-First que hacen al enfoque tan recomendable. (2) las evidencias empíricas de los resultados de API-First, son insuficientes (científicamente), es necesario realizar casos de estudio, estudios de campo, encuestas que aporten datos más precisos y validos sobre las características y propiedades del enfoque, (3) las guías para establecer u orientar un proceso, al ser tan heterogéneas, resultan de limitada utilidad, es necesario considerar el desarrollo de un framework genérico de actividades API-First y/o modelos técnicos/operativos y modelos de gestión / estratégicos, (4) definir con mayor precisión la relación con Agiles, DevOps, gobernanza y administración de APIs, desde las practicas o de automatización, herramientas, métricas, etc. (5) A partir del rol central que adquieren las APIs en todo el proceso de desarrollo de software, describir y/o definir como las APIs se documentan en los diversos artefactos del desarrollo de software (historias

de usuario, diagramas de clase, etc.), (6) debido a que se hace reiterado hincapié en que API-First es una excelente alternativa de negocios para las empresas que lo adopten, éste debe ser demostrado con más datos empíricos, (7) se observan escasos ítems que mencionen desventajas o riesgos del enfoque API-First, es necesario indagar sobre las limitaciones del enfoque más profundamente.

Trabajo Futuro

El trabajo está dirigido a analizar proyectos que incorporen API-First y dar mayor precisión a diversas afirmaciones, ejecutando un estudio confirmatorio. Para ello, se planifica realizar un estudio de campo en GitHub.

VI. REFERENCIAS BIBLIOGRÁFICAS

- [1] R. T. Fielding, «Architectural styles and the design of network-based software architectures,» *Universidad de California, Irvine*, 2000.
- [2] «Apidays. Platformable (2022),» API Landscape State of the Market 2022, [En línea]. Available: <https://apilandscape.apiscene.io/>. [Último acceso: 28 Agosto 2023].
- [3] B. Doerrfeld, C. Wood, A. Anthony, K. Sandoval y A. Lauret, «The API Economy. Disruption and the Business of APIs (1st ed.),» 2016. [En línea]. Available: <https://nordicapis.com/ebooks/the-api-economy/>.
- [4] W. Tan, Y. Fan, A. Ghoneim y M. A. Hossa, «From the Service-Oriented Architecture to the Web API Economy,» vol. 20, n° 4, pp. 64-68, 2016.
- [5] G. Levin, «An API-First Development Approach. An API-First Development Approach,» 15- 11-2016. [En línea]. Available: <https://dzone.com/articles/an-api-first-development-approach-1>. [Último acceso: 18- 8-2024].
- [6] N. Beaulieu, S. Dascalu y E. Hand, «API-First Design: A Survey of the State of Academia and Industry,» de *ITNG 2022 19th International Conference on Information Technology-New Generations* , 2022, pp. (pp.73-79).

- [7] Postman, «State of the API Report,» 2023. [En línea]. Available: <https://voyager.postman.com/pdf/2023-state-of-the-api-report-postman.pdf>. [Último acceso: 19 -10- 2024].
- [8] Postman, «State of the API Report,» 2024. [En línea]. Available: <https://www.postman.com/state-of-api/2024/>. [Último acceso: 24 -10- 2024].
- [9] V. Garousi, M. Felderer y M. Mäntylä, «Guidelines for including grey literature and conducting multivocal literature reviews in software engineering,» *Information and Software Technology*, vol. 106, pp. 101-121, 2019.
- [10] K. Petersen, S. Vakkalanka y L. Kuzniarz, «Guidelines for conducting systematic mapping studies in software engineering: An update,» *Information and software technology*, vol. 64, pp. 1-18, 2015.
- [11] K. Nybom, A. Ashraf y I. Porres, «A Systematic Mapping Study on Tools for API Documentation Generation (Report 1180),» *TUCS Technical Reports*, 2017.
- [12] E. Mosqueira-Rey, D. Alonso-Ríos, V. Moret-Bonillo , I. Fernández-Varela y D. Álvarez-Estévez, «A systematic approach to API usability: Taxonomy-derived criteria and a case study,» *Information and Software Technology*, 97, ., vol. 97, pp. 46-63, 2018.
- [13] R. Boateng, J. Ofoeda y J. Effah, «Application Programming Interface (API) Research: A Review of the Past to Inform the Future,» *International Journal of Enterprise Information Systems (IJEIS)*, vol. 15, no. 3, pp. , Jul. 2019, doi.:, vol. 15, n° 3, p. 76–95, 2019.
- [14] M. Mathijssen, M. Overeem y S. Jansen, «Identification of Practices and Capabilities in API Management: A Systematic Literature Review,» arXiv: Software Engineering., 2020. [En línea]. Available: <https://doi.org/10.48550/arXiv.2006.10481>.
- [15] R. Koçi, J. Franch Gutiérrez, P. Jovanovic y A. Abelló Gamazo, «Classification of Changes in API Evolution,» *IEEE 23rd International Enterprise Distributed Object Computing Conference (EDOC)*, pp. 243-249, 2019.
- [16] S. Casas, D. Cruz, G. Vidal y M. Constanzo, «Uses and applications of the OpenAPI/Swagger specification: a systematic mapping of the literature,» *40th. International Conference of the Chilean Computer Science Society (SCCC)*, <https://doi.org/10.1109/sccc54552.2021.9650408>. 2021.
- [17] M. Siddiqui, «Finextra,» Why Should You Adopt API-first? A Brief Account, 17 -9- 2022. [En línea]. Available: <https://www.finextra.com/blogposting/22918/why-should-you-adopt-api-first-a-brief-account>. [Último acceso: 19 -10- 2024].
- [18] O. Hämäläinen, «API-First Design with Modern Tools. Thesis, School of Business, Degree Programme in Business Information Technology. Jamk University of Applied Sciences,» 2019. [En línea]. Available: <https://urn.fi/URN:NBN:fi:amk-2019060615295>. [Último acceso: 23 -10- 2024].
- [19] J. J. Londoño Tirado, «Gobierno de APIs, implementación y experimentación con API-First y OpenAPI en el proyecto P2P energía transactiva. Tesis de grado. Universidad de Antioquia.,» 2022. [En línea]. Available: <http://hdl.handle.net/10495/29121>. [Último acceso: 20 -10- 2023].
- [20] J. Larsson y L. Åkermark, «The Value Of Implementing API-First As A Methodology When Developing Apis. Dissertation, 2021,» 2021. [En línea]. Available: <https://urn.kb.se/resolve?urn=urn:nbn:se:hj:diva-54311>. [Último acceso: 26 09 2024].
- [21] S. Easterbrook, J. Singer, M.-A. Storey y D. Damian, «Selecting Empirical Methods for Software Engineering Research,» de *Guide to Advanced Empirical Software Engineering*, Londres, Springer, https://doi.org/10.1007/978-1-84800-044-5_11. 2008, p. 285–311.
- [22] M. Shaw, «Writing good software engineering research papers: minitutorial,» *ICSE '03: Proceedings of the 25th International Conference on Software Engineering*, p. 726–736, 2003.
- [23] D. Brajesh, API Management: An Architect's Guide to Developing and Managing APIs for Your Organization (1st ed.), Apress Berkeley, CA, 2017.

VII. ANEXO

Cod	URL - Item
I1	https://apievangelist.com/2014/08/11/what-is-an-api-first-strategy-adding-some-dimensions-to-this-new-question/ https://apievangelist.com/2020/03/09/what-is-api-first/
I2	https://auth0.com/blog/the-business-value-of-api-first-design/
I3	https://blog.stoplight.io/api-first-api-design-first-or-code-first-which-should-you-choose https://blog.stoplight.io/api-first-vs-api-design-first-a-comprehensive-guide
I4	https://blog.treble.com/building-a-node-api-with-api-first-design/
I5	https://bump.sh/blog/dev-guide-api-design-first
I6	https://checkr.com/blog/why-future-thinking-companies-are-betting-on-api-first
I7	https://community.sap.com/t5/technology-blogs-by-sap/what-is-wrong-with-api-first-microservices/ba-p/13353315
I8	https://dev.to/pacheco/api-first-development-theory-and-practice-571d
I9	https://developerexperience.io/articles/api-first-development
I10	https://developers.redhat.com/articles/2022/10/18/api-first-approach-building-nodejs-applications
I11	https://digital.nhs.uk/blog/tech-talk/2021/lessons-from-the-coalface-of-api-first-design
I12	https://dzone.com/articles/what-api-first https://dzone.com/articles/what-is-api-first
I13	https://estafet.com/automated-testing-in-the-early-stages-of-an-api-first-approach-to-api-development/
I14	https://hackernoon.com/is-api-first-development-approach-right-for-your-business-0lr2b58
I15	https://itnext.io/a-complete-guide-to-the-api-first-approach-eed796dd0f10
I16	https://konghq.com/resources/e-book/become-api-first-company https://konghq.com/resources/reports/unlocking-the-api-first-operating-model

I17	https://newsletter.techworld-with-milan.com/p/what-is-api-first-development
I18	https://blogs.cisco.com/tag/api-first https://newsroom.cisco.com/c/r/newsroom/en/us/a/y2022/m10/cisco-advances-api-first-strategy-to-empower-developers-in-the-digital-economy.html
I19	https://nordicapis.com/difference-between-api-first-and-api-as-a-product/ https://nordicapis.com/what-are-the-benefits-of-api-first-development/
I20	https://rapidapi.com/guides/api-first
I21	https://smartbear.com/blog/api-first-development-and-the-case-for-api-mocking/
I22	https://swagger.io/blog/api-design/api-first-approach-with-openapi-training/ https://swagger.io/resources/articles/adopting-an-api-first-approach/
I23	https://thenewstack.io/api-first-development-architecting-applications-with-intention/ https://thenewstack.io/creating-an-api-first-culture-and-company-part-1/
I24	https://tyk.io/blog/res-what-is-api-first/
I25	https://www.adyen.com/knowledge-hub/api-first-adyen
I26	https://www.algolia.com/blog/product/the-5-principles-of-api-first-development-and-what-does-api-first-even-mean/
I27	https://www.apiscene.io/api-business-models/api-first-the-essence-of-digital-transformation/ https://www.apiscene.io/api-business-models/we-should-have-built-an-api-first/
I28	https://www.baeldung.com/spring-boot-openapi-api-first-development
I29	https://www.bairesdev.com/blog/api-first-design-the-ultimate-guide/
I30	https://cloud.google.com/blog/products/api-management/api-first-digital-transformation https://www.googlecloudcommunity.com/qsqph94282/attachments/qsqph94282/cloud-apigee/43255/1/api-first-development-with-swagger-node-1282015.pdf
I31	https://www.computer.org/publications/tech-news/trends/api-first-companies/
I32	https://www.contentful.com/blog/what-is-api-first/

I33	https://www.f5.com/company/blog/nginx/benefits-of-api-first-approach-to-building-microservices https://www.f5.com/glossary/api-first
I34	https://www.finextra.com/blogposting/22918/why-should-you-adopt-api-first-a-brief-account
I35	https://www.forbes.com/sites/forbestechcouncil/2022/07/01/the-five-principles-of-api-first-development-and-what-api-first-really-means/
I36	https://www.ibm.com/docs/en/zos-connect/zos-connect/3.0?topic=concepts-what-is-api-first-development
I37	https://www.infoq.com/articles/api-first-integration/

I38	https://www.mongodb.com/blog/post/mach-aligned-retail-api-first
I39	https://www.postman.com/api-first/#:~:text=API%2Dfirst%2C%20also%20called%20the,of%20treating%20them%20as%20afterthoughts https://www.postman.com/lp/api-first-company/
I40	https://www.redsauce.net/en/article?post=api-first-design
I41	https://www.sanity.io/api-cms
I42	https://www.techtarget.com/searchapparchitecture/tip/Should-you-adopt-an-API-first-approach-to-development

Recibido: 2025-08-16

Aprobado: 2025-10-23

Hipervínculo Permanente: <https://doi.org/10.54789/reddi.10.2.3>

Datos de edición: Vol. 10 - Nro. 2 - Art. 3

Fecha de edición: 2025-12-30

