



Ecosistemas de APIs WEB: Análisis y Caracterización Socio-Técnica

Web API Ecosystems: Socio-Technical Analysis and Characterization

Casas, Sandra^{1*}

Constanzo, Marcela¹

Vidal, Graciela¹

Cruz, Diana¹

¹ Universidad Nacional de la Patagonia Austral. Instituto de Tecnología Aplicada, Grupo de I+D en Ingeniería de Software Pragmática (GISP). Río Gallegos, Argentina

Recibido: 06 Ago. 2024 | **Aceptado:** 30 Oct. 2024 | **Publicado:** 20 Ene. 2025

Autor de correspondencia*: sicasas@uarg.unpa.edu.ar

Cómo citar este artículo: Casas, S., Constanzo, M., Vidal, G., & Cruz, D. (2025). Ecosistemas de APIs WEB: Análisis y Caracterización Socio-Técnica. *Revista Científica de Sistemas e Informática*, 5(1), e757. <https://doi.org/10.51252/rcsi.v5i1.757>

RESUMEN

La transformación digital y la economía API han dado origen a ecosistemas digitales, donde las APIs Web son los activos principales. Los Ecosistemas de APIs Web (EAW) reúnen en una plataforma a personas, empresas, servicios, recursos, herramientas, innovadoras actividades y procesos de desarrollo. En el ámbito de la Ingeniería de Software, los EAW son Ecosistemas de Software (SECO). Una forma de examinar y entender los sistemas complejos es el enfoque Socio-Técnico (S-T), el cual disminuye el riesgo de que los sistemas no cumplan los objetivos organizacionales. Según diversos estudios de revisión publicados, no se registran estudios S-T en relación con las API web como en SECO. Con el objetivo de identificar categorías de análisis, ejecutamos un caso de estudio exploratorio y descriptivo aplicado a cuatro EAW actuales, aplicando un enfoque S-T. Encontramos y clasificamos los componentes de las dimensiones social, infraestructura y normas en cinco categorías principales, las cuales además incluyen subcategorías. Las relaciones con el contexto se han clasificado en tres categorías. Finalmente, discutimos y comparamos nuestras observaciones con la literatura más difundida respecto a la gestión de API y los Ecosistemas de Software.

Palabras clave: administración de API; API Web; ecosistema de software; sistema socio-técnico

ABSTRACT

Digital transformation and the API economy have given rise to digital ecosystems, where Web APIs are the main assets. Web API Ecosystems (WEAs) bring together people, companies, services, resources, tools, and innovative development activities and processes on a platform. In the field of Software Engineering, WEAs are Software Ecosystems (SECOs). One way to examine and understand complex systems is the Socio-Technical (S-T) approach, which reduces the risk of systems not meeting organizational objectives. According to various published review studies, there are no S-T studies reported in relation to Web APIs as in SECO. In order to identify categories of analysis, we performed an exploratory and descriptive case study applied to four current WEAs, applying an S-T approach. We found and classified the components of the social, infrastructure, and standards dimensions into five main categories, which also include subcategories. Relationships with the context have been classified into three categories. Finally, we discuss and compare our observations with the most widely disseminated literature regarding API Management and Software Ecosystems.

Keywords: API management; Web API; software ecosystem; socio-technical system



1. INTRODUCCIÓN

Desde la creación de la arquitectura REST en el año 2000 (Fielding & Taylor, 2000), la “apificación” (adopción de APIs web) ha sido continua. Las APIs web constituyen el pilar del software moderno al permitir conectar aplicaciones software usando la red. Las organizaciones de todo tipo (gobiernos, empresas y negocios) conectan digitalmente sus aplicaciones a interfaces de usuario accesibles para clientes, empleados y usuarios remotos como el principal medio de interacción. Las APIs web son quizás una de las más claras formas de la amplitud de la globalización del desarrollo de software.

La Economía API (Tan et al., 2016) y la Transformación Digital han impulsado un mercado de productos digitales y una industria que traza innovación y que avanza a pasos agigantados, en torno a las APIs web. Este sector (mercado+industria) comprende productos que van desde las mismas APIs, hasta herramientas para soportar el ciclo de vida de las APIs (diseño, prueba, etc.), soluciones de gestión de API, plataformas de documentación de API, herramientas de seguridad de API, paneles de control de API, etc. La industria evolucionó de productos API a plataformas y de empresas de todo tamaño a ecosistemas digitales (Apidays, 2022).

La investigación científica de APIs web en áreas del desarrollo del software y sistemas distribuidos, se ha centrado en problemas y soluciones de carácter técnico y/o tecnológico, según Ofoeda et al. (2019). Los mismos, señalan que la investigación es atórica y recomiendan que la investigación de APIs web también debe considerarse desde una perspectiva Socio-Técnico (S-T). A su vez, los autores Raatikainen et al. (2021) indican que aún la literatura científica no realiza suficiente distinción entre APIs estáticas y APIs web, y critican que la investigación a menudo propone pautas o prácticas en lugar de caracterizar el estado de la práctica industrial en términos de desafíos y soluciones probadas.

Debido al impacto de las Plataformas Digitales o Ecosistemas de Plataformas (PE) en la oferta de productos y servicios, la competencia entre las empresas, la generación de nuevos negocios desde la economía y la innovación es que el estudio de estas, desde diversas perspectivas, es considerado un tópico de interés multidisciplinar (Kapoor et al., 2021). A su vez, los Ecosistemas de Software (SECO) son un tema emergente de estudio en la Ingeniería de Software (Manikas & Hansen, 2013). La indagación de SECO enfrenta desafíos diversos, donde los aspectos técnicos del desarrollo de software se mezclan con cuestiones sociales. En este contexto, el modelado socio-técnico también se considera esencial, aunque sigue siendo una tarea pendiente (Jansen et al., 2015).

Según Baxter & Sommerville (2011) la motivación detrás de la aplicación de la perspectiva S-T surge de la idea de que no hacerlo aumenta el riesgo de que un sistema no haga las contribuciones esperadas hacia los objetivos organizacionales. Una mirada técnica ofrece información sobre cómo un sistema cumple con los requisitos técnicos, pero no tiene en cuenta las intrincadas relaciones entre una organización y los actores que emprenden y respaldan los diversos procesos de negocio (Baxter & Sommerville, 2011). Para establecer una comprensión coherente de cómo el sistema social (actores de la red) ajusta/redefine los objetivos para respaldar los aspectos técnicos (cambios técnicos e innovaciones dentro del PE), se considera adecuado un enfoque S-T. Los sistemas S-T son inherentemente dinámicos y evolucionan a través de la configuración recursiva de infraestructura técnica y construcciones sociales, lo que se refleja en acciones que alteran entidades a nivel técnico, de tarea, de estructura y de actores (Dremel et al., 2020).

Los Ecosistemas de API web (EAW) son entramados complejos y dinámicos de componentes sociales (actores) y técnicos (herramientas) que, bajo un sistema de normas, evolucionan con innovaciones para el desarrollo de software.

En este trabajo presentamos un estudio de caso exploratorio y descriptivo sobre cuatro EAW actuales, con el objetivo de identificar categorías de análisis que profundizan nuestros hallazgos previos (Casas et al., 2023), desde un enfoque S-T. Finalmente discutimos las categorías de análisis en relación a la bibliografía relacionada a la administración de APIs y posibles aportes al modelado de SECO.

Trabajos relacionados

Este estudio se enmarca en varios contextos teóricos, APIs web, administración de APIs, Ecosistemas de Software, Ecosistemas de Plataforma y Sistemas S-T.

Desde las áreas de la ingeniería de software y sistemas distribuidos, las APIs web son objeto de estudio hace dos décadas, según diversas revisiones los temas dominantes de la investigación hacen eje en los aspectos tecnológicos como la usabilidad (Mosqueira-Rey et al., 2018), evolución (Koci et al., 2019), documentación (Nybom et al., 2017; Cummaudo et al., 2019), especificación (Casas et al., 2021) y desarrollo (Beaulieu et al., 2022). En particular, la revisión sistemática de la literatura publicada entre 2010 y 2018 de Ofoeda et al. (2019), que comprende 104 artículos, presenta una descripción general de la investigación de API, sintetiza la investigación del periodo y describe las áreas que necesitan mayor atención. Los hallazgos sugieren que los temas dominantes en la investigación de APIs web pertenecen al diseño y la usabilidad, que hacen foco al dominio tecnológico del esquema de clasificación. El estudio también señala que la investigación de API es en gran parte atórica. La mayoría de los estudios tienden a utilizar experimentos y muestras de código como métodos de investigación frente a los métodos cualitativos y cuantitativos dominantes. Los autores identifican como desafío para la investigación de API el abordaje desde una perspectiva S-T, como fue indicado en Casas et al. (2021).

La administración o gestión de APIs, es también un tópico incipiente. Una referencia clásica para profesionales e investigadores de APIs web es De (2017), que trata la administración o gestión de API. El autor, se refiere por lo general a plataforma de administración de servicios API o plataforma de empresa API. Aporta los siguientes conceptos, “Una plataforma de administración de API ayuda a una organización a publicar APIs para desarrolladores internos, asociados, externos y exponer el potencial de sus activos. Brinda las capacidades básicas para garantizar un programa de API exitoso a través de la participación de los desarrolladores, conocimientos comerciales, análisis, seguridad y protección. Una plataforma de administración de APIs ayuda a las empresas a acelerar el alcance a través de los canales digitales, impulsar la adopción de socios, monetizar activos digitales y proporcionar análisis para optimizar las inversiones en transformación digital”. El texto realiza una completa descripción y análisis desde el punto de vista técnico/tecnológico (diseño, documentación, pruebas, seguridad, control de versiones de APIs, analíticas y gobierno de APIs, monetización, portal del desarrollador).

Mathijssen et al. (2020) presenta una revisión sistemática de la literatura, que tiene el objetivo de recopilar prácticas y capacidades de gestión de API, así como proponer una definición integral del tema. Los autores recopilan 24 definiciones únicas para el tema, 114 prácticas y 39 capacidades.

Con una orientación en las disciplinas de negocios, comercio y empresarial, Kapoor et al. (2021) revisa y analiza la literatura publicada entre 1999-2019 en relación a Ecosistemas de Plataformas, (PE) desde un enfoque S-T. Señala como uno de los resultados más relevantes, que los estudios existentes están interesados principalmente en el aspecto técnico de las plataformas y otros aspectos tangibles (componentes y recursos) y menos en el aspecto social.

Manikas & Hansen (2013) revisaron la literatura sobre ecosistemas de software evidenciando un crecimiento en el interés sobre su estudio. Además, notaron que empezaban a surgir estudios con perspectivas social y empresarial, a parte de la perspectiva técnica. Una primera definición que apareció, desde una perspectiva técnica, fue formulada por Messerschmitt & Szyperski (2003), como un conjunto de empresas que funcionan como una unidad e interactúan con un mercado compartido de software y servicios, relacionándose entre ellos.

La complejidad de las relaciones entre elementos del ecosistema de software requiere técnicas analíticas para predecir los efectos de las decisiones estratégicas sobre la red de proveedores, clientes, colaboradores y el ecosistema (McGregor, 2012). Estos desafíos requieren analizar el campo de los ecosistemas de softwares desde otra perspectiva. En este sentido, el estudio de los ecosistemas de software es un tópico

emergente, que investiga las relaciones entre las empresas de la industria del software (Pereira dos Santos et al., 2012). Por lo general, estas relaciones están respaldadas por una plataforma tecnológica común y operan a través del intercambio de información, recursos y artefactos.

A pesar de los avances iniciales en la investigación de SECO evidenciados en diversos estudios de mapeo o revisión sistemática, según Manikas (2016), son insuficientes los modelos analíticos, estudios de casos con datos reales y soporte de herramientas integradas. Un gran obstáculo para el avance con aportes concretos para la industria real es la falta de soporte de modelado SECO. De acuerdo con Jansen et al. (2015), el modelado SECO es importante para proporcionar información a partir de representaciones que permiten analizar y comparar, ecosistemas "estáticos", basados en conceptos clave (por ejemplo, organizaciones, relaciones y flujos) y métodos existentes (por ejemplo, redes socio-técnicas y redes de suministro de software). Algunas propuestas de modelado de SECO, son presentadas por Campbell & Ahmed (2010), Boucharas et al. (2009), Pereira dos Santos & Werner (2011, 2012) que se detallan a continuación.

Campbell & Ahmed (2010) propusieron analizar a las SECO en tres dimensiones: dimensión empresarial (conocimiento del mercado, y el papel clave de los tomadores de decisiones que identifican las necesidades de SECO y las expansiones de la plataforma, y mantienen el portafolio de productos), dimensión arquitectónica (la definición y mantenimiento de las tecnologías de las plataformas, es decir, lo necesario para mejorar la calidad, la interoperabilidad entre sistemas y las evaluaciones de desempeño son algunos de los aspectos comúnmente analizados en la plataforma) y dimensión social (conocimiento sobre los diversos actores que desempeñan distintos roles como usuarios, clientes, proveedores y desarrolladores, interactuando entre ellos). Boucharas et al. (2009) propusieron una perspectiva de modelado de SECO basada en tres niveles: Primer nivel: Es el nivel de alcance organizacional, donde los objetos de estudio son los actores y sus relaciones en el contexto de la organización perteneciente a alguna SECO. Segundo nivel: Las SSN (Software Supply Network) y sus diferentes relaciones son el objeto de estudio. El foco está en las conexiones entre software, hardware y servicios organizacionales. Tercer nivel: Los objetos de estudio son las propias SECO, incluyendo las relaciones entre ellas.

Es importante que las SECO tengan sus límites bien definidos como una tecnología, una plataforma, un mercado o una empresa. Además de tomar en cuenta el ciclo de vida de los ecosistemas, Pereira dos Santos & Werner (2011) y Pereira dos Santos & Lima Werner (2012) dividen las actividades que componen el proceso de evolución de una aplicación a un ecosistema en "3+1" dimensiones: (i) "dimensión arquitectónica (o técnica)" – centrada en la plataforma SECO (es decir, mercado, tecnología, infraestructura u organización) – (ii) "dimensión negocio (o transacciones)" centrada en el flujo de conocimiento de SECO (es decir, artefactos, recursos e información); (iii) "dimensión social" – centrada en las partes interesadas (stakeholders) de la SECO y (iv) "dimensión Ingeniería y gestión en la SECO, que fusiona las tres dimensiones básicas y abarca actividades relacionadas con la economía.

Lima et al. (2015) representan un ecosistema de software como un entorno en el que el repositorio de componentes reúne a las partes interesadas y los componentes de software. Esos repositorios necesitan recursos S-T para dar soporte a las relaciones de los actores.

En Casas et al. (2023) se presenta el primer estudio S-T de Ecosistemas de APIs Web, cuyo objetivo fue comprender la estructura y componentes de los mismos. El mismo establece premisas, actividades, dimensiones y contexto de los EAW. El estudio implicó la construcción de un framework S-T para analizar tres EAW. Este instrumento se elaboró a partir de los enfoques y aportes específicos de sistemas S-T de Kingdon (1995), Baxter & Sommerville (2011) y Elatlassi et al. (2022).

2. MATERIALES Y MÉTODOS

El estudio cualitativo que se presenta consiste en un caso de estudio exploratorio y descriptivo sobre cuatro EAW actuales, usando las directrices de Creswell (2014), Fabbri (2020) y Runeson et al. (2012). A partir

de la formulación de categorías de análisis sobre los elementos S-T, el objetivo es profundizar la comprensión, conceptualización y caracterización de los componentes S-T de los EAW. La pregunta que se formula refiere ¿cómo se categorizan los componentes S-T de los EAW? La Figura 1 grafica el diseño metodológico aplicado. A continuación, describimos el framework y luego cómo se organizó su aplicación.

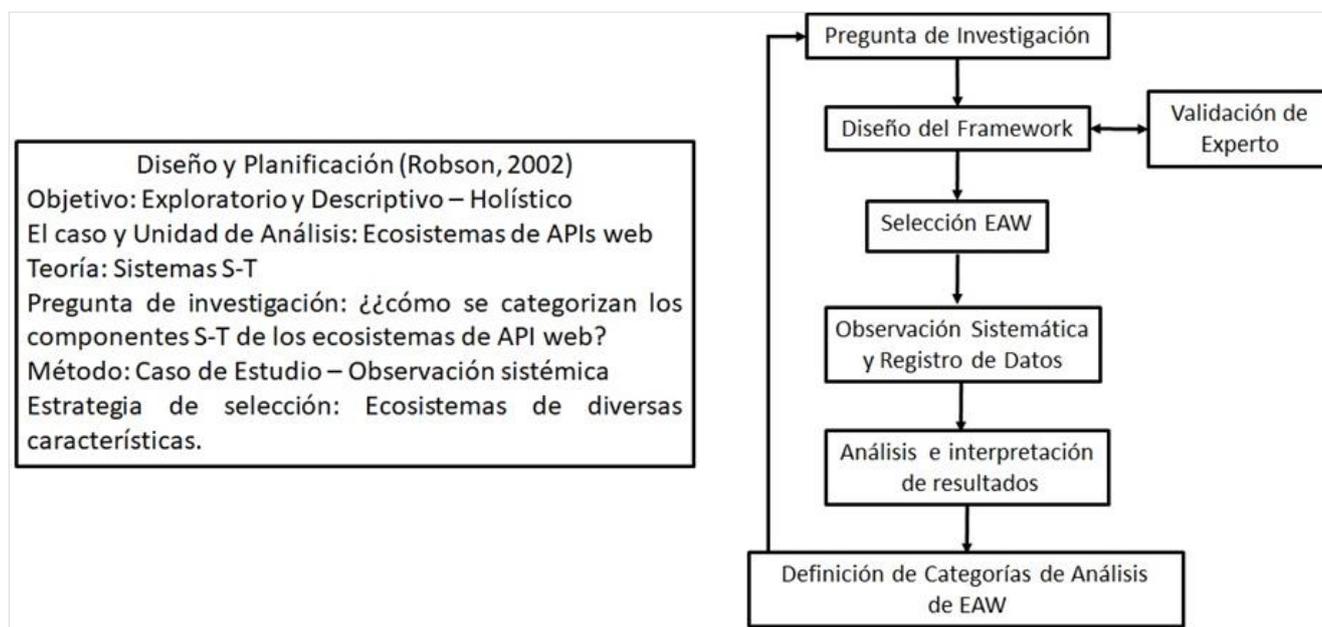


Figura 1. Diseño Metodológico

2.1. Framework

Esta herramienta fue elaborada y presentada por Casas et al. (2023). La Figura 2 representa la estructura general, que a continuación se describe brevemente.

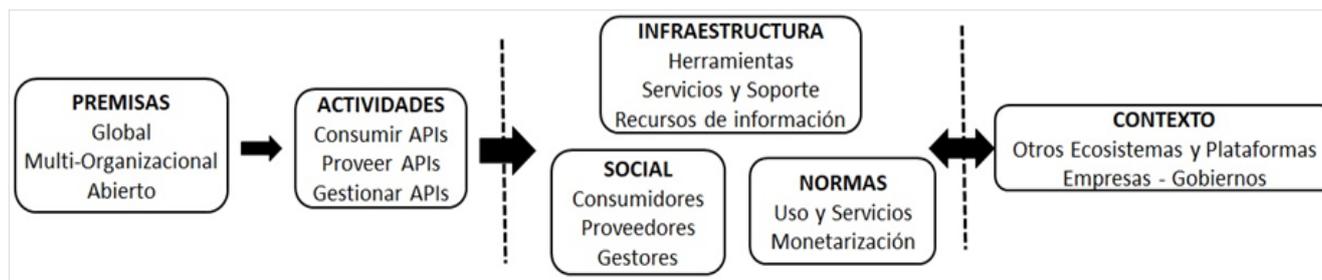


Figura 2. Framework S-T

- a) Premisas: Conjunto de características distintivas que un sistema o PE debe cumplir para ser considerado un ecosistema de API web.
 - Global: Se establece en torno a una plataforma digital.
 - Multi-organizacional: Participan activamente en el ecosistema diversas organizaciones (empresas, gobiernos, etc.).
 - Abierto: Puede participar del ecosistema cualquier persona, equipo u organización en tanto acepte y cumpla las normas establecidas.
- b) Actividades: Las actividades son, consumir APIs web y producir APIs web, y gestionar la administración de estas en la plataforma.
- c) Dimensión Social: Personas o empresas que participan o son parte del ecosistema.
- d) Dimensión Infraestructura: Conjunto de herramientas, recursos y servicios ofrecidos desde la plataforma para consumir o proveer APIs web.

- e) Dimensión Normas: Conjunto de normas (reglas, pautas y políticas) que se establecen para regular las relaciones y acciones de los diversos actores del ecosistema. Refieren a los compromisos que asumen los consumidores y proveedores, así como los gestores.
- f) Contexto: Ecosistemas, plataformas, sistemas y/o empresas con las que el EAW establece relaciones y no forman parte de los conjuntos de consumidores o proveedores de APIs.

2.2. Recolección de datos

La selección de los ecosistemas RapidAPI, APILayer, Nubentos y Apilanding muestran características bien diferenciadas, en cuanto a, permanencia y posición en el mercado (nivel de madurez y consolidación), cantidad de usuarios (consumidores y proveedores), cantidad de APIs y dominio. El proceso de aplicación del framework a los EAW mencionados se realizó durante los años 2022 (RapidAPI, APILayer, Nubentos) y 2023 (Apilanding).

Las técnicas de observación sistemática o estructurada según Creswell (2014) y Fabbri (2020) se usaron para la recopilación de los datos. Un formulario basado en los componentes del framework, se utilizó para registrar las observaciones y experiencias. Se indagaron e inspeccionaron todos los contenidos (documentación, videos, formularios, plantillas, blogs, etc.) de cada plataforma correspondiente a los EAW. A partir del registro como usuarios se probaron las herramientas y asistentes, entre otras acciones que se hicieron para recabar todos los datos de los EAW. Luego de obtenidos estos datos, la información se unificó, procesó, y analizó manualmente para proceder a la definición de las categorías de análisis.

Se siguieron las indicaciones de Creswell (2014) en el diseño y ejecución del caso de estudio, para minimizar las amenazas a la validez, sesgos y limitaciones, con relación a la validez descriptiva (interna), la validez externa y la generalización de los resultados.

3. RESULTADOS

3.1 Categorías de análisis de los ecosistemas de APIs Web

La Figura 3 esquematiza las categorías de análisis determinadas a partir de la ejecución del caso de estudio. A continuación, se presentan detalles de cada ítem de análisis y en el anexo los datos más pormenorizados.

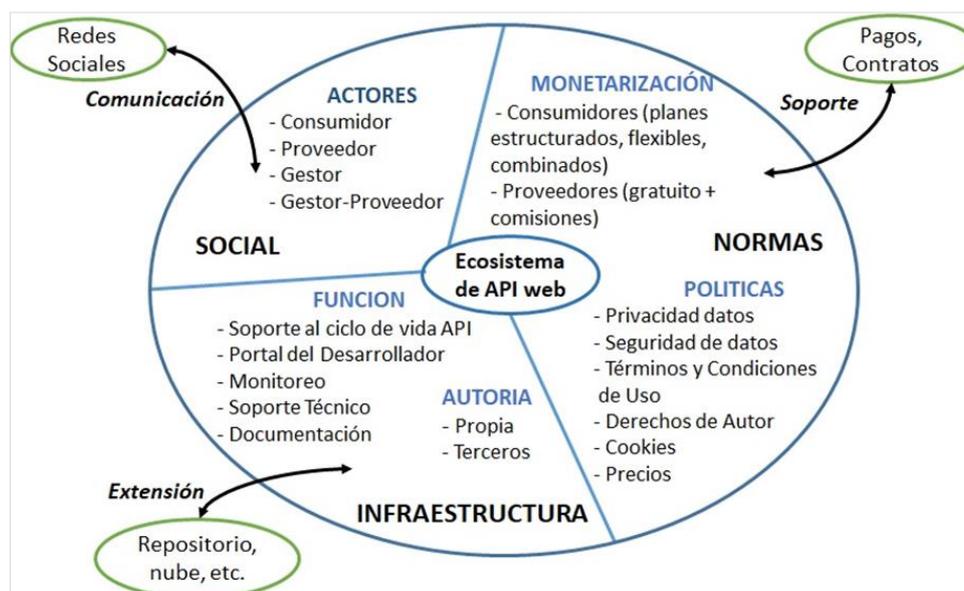


Figura 3. Categorías de Análisis S-T de EAW

- Premisas.- Los cuatro EAW cumplen con las premisas definidas.

- (i) Globales: Se establecen y se acceden a partir de plataformas digitales (<https://rapidapi.com/>, <https://www.nubentos.com/en/apistore>, <https://apilayer.com/> y <https://apilanding.com/web/index.php>).
- (ii) Multi-organizacional: En todos se destaca la participación de diversas empresas e individuos (proveedores, consumidores, del contexto).
- (iii) Abiertos: Fue constatado a partir de la creación de cuentas y uso de las diversas funcionalidades, tanto como consumidores y proveedores de APIs.

La Tabla 1 presenta características generales de los mismos.

Tabla 1. Características generales de las EAWs analizadas

EAW	Origen	Dominio	Cantidad de APIs	Desarrolladores
RapidAPI	USA 2015	General	Más de 40000	Más de 4000000
APILayer	USA	General	Más de 80	1200000
Nubentos	España-2018	Salud	Menciona 30 proveedores	Menciona 10 empresas consumidoras
Apilanding	Argentina	Finanzas	15 propias + 218 de terceros	Menciona algunas empresas clientes de renombre.

- **Actividades.**- Los cuatro EAW ofrecen catálogos con APIs web de diversos proveedores que pueden ser consumidas. La administración de las APIs y la plataforma posibilita las mismas.

- **Dimensión Social.**- En todos los EAW se distinguen los tres actores, consumidores, proveedores y gestores. En el caso de APILayer y Apilanding cumplen dos roles, dado que además de ser los gestores también son proveedores, pero no son los únicos. Se observa que, en los casos de doble rol, el ecosistema pertenece a una empresa IT que ofrece otros productos y servicios, en el caso de APILayer es propiedad de Idera Inc. y Apilanding es de la empresa Sysworld Servicios S.A., mientras que Nubentos y RapidAPI son organizaciones que solo se dedican a este negocio. Identificamos cuatro categorías de actores: consumidor, proveedor, gestor y gestor-proveedor.

- **Dimensión Infraestructura.**- En todos los EAW se proporcionan diversas herramientas, servicios, recursos. Se han categorizado por la función que cumplen y la autoría de estas.

- **Función.**- Las herramientas y recursos se clasifican en cinco agrupamientos.

- (i) Herramientas para soportar el ciclo de vida de API, destinadas a proveedores, permiten publicar, diseñar, probar, manejar versiones, de las APIs.
- (ii) Herramientas para consumir las APIs, destinadas a consumidores, en general son SDK, CLI, editores en línea, herramientas de pruebas, generadores de código en diversos lenguajes que permiten integrar las APIs.
- (iii) Herramientas y servicios de soporte (técnico), destinadas tanto a consumidores como proveedores, con facilidades como generadores de tickets, chat en línea, formularios de problemas y/o errores.
- (iv) Recursos de Documentación, destinados tanto a consumidores como proveedores y refiere a todo recurso (tutoriales, libros, artículos, blogs, ejemplos, videos, get started, preguntas frecuentes, etc.) que guía y ayuda a consumir o a proveer una API.
- (v) Servicios de monitoreo, consiste en aquella información (métricas e indicadores) relacionada al desempeño de las APIs y/o la puerta de enlace (plataforma).

- **Autoría.**- El desarrollo o construcción de las principales herramientas puede ser:

- (i) Propia: La empresa que gestiona y “dueña” de la plataforma es la desarrolladora de las principales herramientas de la plataforma. Este sería el caso de RapidAPI, Apilanding y Nubentos.

- (ii) De terceros: la mayor parte de las herramientas para dar soporte a las actividades del ecosistema son desarrollos externos o de terceros, este sería el caso de APILayer.

- Dimensión Normas.- Como otras plataformas digitales o ecosistemas de plataformas, en la plataforma se publican contenidos (texto, imágenes, videos) que son propiedad de los gestores, proveedores y/o consumidores. El uso de esta información, los otros recursos/servicios debe ser legal, autorizado y controlado. Condiciones y términos de uso y servicio, derechos de autor, confidencialidad de datos, seguridad, entre otros, sigue el esquema de la mayoría de las plataformas o sitios web. Las categorías que se distinguen son:

- (i) Políticas de Privacidad de Confidencialidad/datos.
- (ii) Políticas de Seguridad de datos.
- (iii) Términos y Condiciones de Uso (Servicios/Plataforma).
- (iv) Derechos de Autor.
- (v) Política de Cookies.
- (vi) Política de Precios.

Todos los EAW presentan un modelo de monetización. Los contratos con los consumidores, se definen en torno a los servicios/productos que se ofrecen y un esquema basado en planes, según las necesidades de los clientes. Los productos y servicios, son una combinación del uso de la plataforma, atención y soporte, y consumo de las APIs. Sin embargo, se ofrecen alternativas que flexibilizan el esquema. Se observan:

- (i) Planes estructurados.
- (ii) Planes flexibles.
- (iii) Combinación de planes.

Otra cuestión del modelo, es el aspecto relacionado a los proveedores. La publicación de APIs en los catálogos es gratuita, y en algunos casos se aplican comisiones sobre el consumo de las APIs por parte de los consumidores. En el caso de APILayer el alojamiento en la nube es variable en cuanto a los precios, y en RapidAPI el uso de los productos.

- Contexto.- Los cuatro ecosistemas se relacionan con el contexto con diversos objetivos y en relación con las dimensiones definidas, entre ellos los clasificamos en:

- (i) Para establecer comunicación con actuales o potenciales consumidores y/o proveedores: Todos los ecosistemas se conectan con cuentas en varias redes sociales.
- (ii) Para mejorar o extender las capacidades de la infraestructura: Se utilizan repositorios y herramientas de integración, analíticas, alojamiento en la nube, especificación, diseño y prueba de APIs, entre otras.
- (iii) Para dar soporte a las normas: Se usan plataformas o herramientas para la suscripción de contratos y/o la administración de los cobros/pagos.

4. DISCUSIÓN

A partir del caso de los resultados obtenidos y nuestra interpretación de estos, observamos que:

- a) Las categorías que hemos definido presentan diferencia con otras clasificaciones. En la dimensión Social se han identificado cuatro roles, consumidor, proveedor, gestor y gestor-proveedor. No obstante, De (2017) solo reconoce a consumidor y gestor-proveedor (sin presencia de otros proveedores). Nuestra clasificación de actores se asemeja a la aportada por Kapoor et al. (2021), con diferentes etiquetas (líder para gestor, complementador para proveedor y usuario final para consumidor). Esta diferencia entre los EAW que hemos analizado y De (2017) además define que corresponden a distintos tipos de EP, según la clasificación de Kapoor et al. (2021). Siendo que los conceptos de De (2017) corresponden a plataformas internas, mientras que los EAW RapidAPI, APILayer, Nubentos y Apilanding serían EP

externas. La conformación de estas categorías, también incide en las interacciones entre los grupos de actores, que desencadenan externalidades de red (efectos de red) del mismo lado o cruzadas entre lados, que influyen en el rendimiento, el éxito y la sostenibilidad general del EP una plataforma en comparación con los competidores.

- b) La “comunidad” es un actor social para De (2017) y Mathijssen et al. (2020), que se conforma por el conjunto de consumidores y productores de APIs, los cuales interactúan en foros y blogs en la plataforma. Incluye a la “comunidad” como un actor social e indica que los foros de debate y los blogs que describen las experiencias de los desarrolladores ayudan a crear una comunidad de desarrolladores comprometida. En el estudio realizado a las cuatro plataformas de las EAW, no se hallaron foros para los actores que forman la comunidad. El actor social “comunidad” tampoco es identificado por Kapoor et al. (2021).
- c) Otra diferencia según De (2017) se aprecia en el modelo de monetización, que es más restringido y solo se enfoca en el aspecto de los consumidores.

En síntesis, interpretamos que las diferencias según De (2017) y Mathijssen et al. (2020) se deben, a que la Administración de APIs está concebida como una actividad que genera un ecosistema más acotado, con menor presencia de otros proveedores y la centralidad del gestor con doble rol.

En relación al ámbito de ecosistema de software, Pereira dos Santos et al. (2012) propone estudiar las relaciones entre las empresas de la industria del software que participan. En este trabajo hemos identificado algunas de estas relaciones, en primer lugar, los roles de los principales actores establecen relaciones. Luego, cuando analizamos el contexto y clasificamos a las diversas empresas en función de los objetivos que complementan el EAW. Empero, estas relaciones deben ser analizadas con mayor nivel de profundidad. De manera complementaria, consideramos que los resultados de este estudio contribuyen a la construcción de algunos modelos SECO existentes, ya que pueden mapearse parcialmente algunos elementos/dimensiones. Por ejemplo, las tres dimensiones de Campbell & Ahmed (2010) negocios, arquitectura y social tienen puntos de coincidencia con nuestro enfoque, en cuanto a Boucharas et al. (2009), la definición de los niveles 1 (organizacional) y 3 (relación con otras SECO) puede beneficiarse de nuestros aportes, y por último en relación a Pereira dos Santos et al. (2012), las 4 dimensiones (arquitectónica, negocios, social e ingeniería y gestión) están presentes en nuestra propuesta, la principal diferencia es que no modelamos la evolución de la SECO, sino sus aspectos S-T más relevantes. Por otro lado, Lima et al. (2015) estudia los recursos S-T para dar soporte a las relaciones de los actores, observamos que varios de estos están presentes en la dimensión infraestructura de nuestro análisis, pero a la vez, también observamos la ausencia de otros artefactos.

CONCLUSIONES

A partir del vacío de conocimiento, en cuanto a la necesidad de estudiar las APIs web con un enfoque S-T y la importancia de la aplicación de esta perspectiva en la Ingeniería de Software, presentamos una caracterización S-T basada en categorías de análisis de los EAW. La misma es una profundización conceptual de los ecosistemas de APIs web, ya que se identificaron y describieron las categorías función y autoría de las herramientas para la dimensión Infraestructura, actores para la dimensión social, y políticas y monetización en la dimensión normas. En cuanto al contexto con el cual se relaciona un EAW se han distinguido tres categorías basadas en los objetivos de esta, siendo para establecer comunicación con actuales o potenciales actores, mejorar o extender la infraestructura y para dar soporte a las normas. Esta herramienta facilita la comprensión de estas innovaciones en la industria del desarrollo del software y la Ingeniería de Software. Sin abandonar el enfoque S-T y multidisciplinar, y en observación directa de la industria de APIs web, continuaremos el estudio y análisis de EAW con una mayor orientación a ecosistemas de software.

FINANCIAMIENTO

Los autores no recibieron ningún patrocinio para llevar a cabo este estudio-artículo.

CONFLICTO DE INTERESES

Los autores declaran no tener ningún tipo de conflicto de interés relacionado con el desarrollo del estudio.

CONTRIBUCIÓN DE LOS AUTORES

Conceptualización; Curación de datos; Análisis formal; Investigación; Metodología; Visualización; Redacción - borrador original; Redacción - revisión y edición: Casas, S., Constanzo, M., Vidal, G. y Cruz, D.

REFERENCIAS BIBLIOGRÁFICAS

- Apidays. (2022). *Platformable API Landscape State of the Market 2022*. The API Landscape. <https://apilandscape.apiscene.io/>
- Baxter, G., & Sommerville, I. (2011). Socio-technical systems: From design methods to systems engineering. *Interacting with Computers*, 23(1), 4-17. <https://doi.org/10.1016/j.intcom.2010.07.003>
- Beaulieu, N., Dascalu, S. M., & Hand, E. (2022). *API-First Design: A Survey of the State of Academia and Industry* (pp. 73-79). https://doi.org/10.1007/978-3-030-97652-1_10
- Boucharas, V., Jansen, S., & Brinkkemper, S. (2009). Formalizing software ecosystem modeling. *Proceedings of the 1st international workshop on Open component ecosystems*, 41-50. <https://doi.org/10.1145/1595800.1595807>
- Campbell, P. R. J., & Ahmed, F. (2010). A three-dimensional view of software ecosystems. *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume*, 81-84. <https://doi.org/10.1145/1842752.1842774>
- Casas, S., Constanzo, M., Vidal, G., & Cruz, D. (2023). Ecosistemas de APIs web como un sistema socio-técnico: Un caso de estudio. *Ciencia Latina Revista Científica Multidisciplinar*, 7(1), 9095-9120. https://doi.org/10.37811/cl_rcm.v7i1.5039
- Casas, S., Cruz, D., Vidal, G., & Constanzo, M. (2021). Uses and applications of the OpenAPI/Swagger specification: a systematic mapping of the literature. *2021 40th International Conference of the Chilean Computer Science Society (SCCC)*, 1-8. <https://doi.org/10.1109/SCCC54552.2021.9650408>
- Creswell, J. W. (2014). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (1.^a ed.). SAGE.
- Cummaudo, A., Vasa, R., & Grundy, J. (2019). What should I document? A preliminary systematic mapping study into API documentation knowledge. *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, 1-6. <https://doi.org/10.1109/ESEM.2019.8870148>
- De, B. (2017). *API Management: An Architect's Guide to Developing and Managing APIs for Your Organization* (1.^a ed.). Apress.
- Dremel, C., Herterich, M. M., Wulf, J., & vom Brocke, J. (2020). Actualizing big data analytics affordances: A revelatory case study. *Information & Management*, 57(1), 103121. <https://doi.org/10.1016/j.im.2018.10.007>
- Elatlassi, R., Narwankar, C., & Calvo-Amodio, J. (2022). A Categorization of Socio-Technical Systems Approaches based on Context and Purpose. *International Society for the Systems Sciences*, 16(1), 1-23. <https://journals.iss.org/index.php/proceedings60th/article/view/2982>

- Fabbri, M. S. (2020). *Las técnicas de investigación: la observación* (pp. 1-10).
https://www.academia.edu/36157300/Las_técnicas_de_investigación_la_observación
- Fielding, R. T., & Taylor, R. N. (2000). *Architectural styles and the design of network-based software architectures* [University of California, Irvine]. <https://dl.acm.org/doi/10.5555/932295#cited-by-sec>
- Jansen, S., Handoyo, E., & Alves, C. (2015). Scientists' Needs in Modelling Software Ecosystems. *Proceedings of the 2015 European Conference on Software Architecture Workshops*, 1-6.
<https://doi.org/10.1145/2797433.2797479>
- Kapoor, K., Ziaee Bigdeli, A., Dwivedi, Y. K., Schroeder, A., Beltagui, A., & Baines, T. (2021). A socio-technical view of platform ecosystems: Systematic review and research agenda. *Journal of Business Research*, 128, 94-108. <https://doi.org/10.1016/j.jbusres.2021.01.060>
- Kingdon, J. W. (1995). *Agendas, Alternatives, and Public Policies* (2.^a ed.). HarperCollins College Publishers.
- Koci, R., Franch, X., Jovanovic, P., & Abello, A. (2019). Classification of Changes in API Evolution. *2019 IEEE 23rd International Enterprise Distributed Object Computing Conference (EDOC)*, 243-249.
<https://doi.org/10.1109/EDOC.2019.00037>
- Lima, T., Pereira dos Santos, R., & Lima Werner, C. (2015). A survey on socio-technical resources for software ecosystems. *Proceedings of the 7th International Conference on Management of computational and collective intelligence in Digital EcoSystems*, 72-79.
<https://doi.org/10.1145/2857218.2857230>
- Manikas, K. (2016). Revisiting software ecosystems Research: A longitudinal literature study. *Journal of Systems and Software*, 117, 84-103. <https://doi.org/10.1016/j.jss.2016.02.003>
- Manikas, K., & Hansen, K. M. (2013). Software ecosystems – A systematic literature review. *Journal of Systems and Software*, 86(5), 1294-1306. <https://doi.org/10.1016/j.jss.2012.12.026>
- Mathijssen, M., Overeem, M., & Jansen, S. (2020). Identification of Practices and Capabilities in API Management: A Systematic Literature Review. *Software Engineering*, 1.
<https://arxiv.org/abs/2006.10481>
- McGregor, J. D. (2012). Ecosystem modeling and analysis. *Proceedings of the 16th International Software Product Line Conference - Volume 2*, 268-268. <https://doi.org/10.1145/2364412.2364463>
- Messerschmitt, D. G., & Szyperski, C. (2003). *Software Ecosystem: Understanding an Indispensable Technology and Industry*. The MIT Press. <https://doi.org/10.7551/mitpress/6323.001.0001>
- Mosqueira-Rey, E., Alonso-Ríos, D., Moret-Bonillo, V., Fernández-Varela, I., & Álvarez-Estévez, D. (2018). A systematic approach to API usability: Taxonomy-derived criteria and a case study. *Information and Software Technology*, 97, 46-63. <https://doi.org/10.1016/j.infsof.2017.12.010>
- Nybom, K., Ashraf, A., & Porres, I. (2017). A Systematic Mapping Study on Tools for API Documentation Generation. *Turku Centre for Computer Science*, 1180, 1-22.
<https://doi.org/10.13140/RG.2.2.31115.49444>
- Ofoeda, J., Boateng, R., & Effah, J. (2019). Application Programming Interface (API) Research. *International Journal of Enterprise Information Systems*, 15(3), 76-95. <https://doi.org/10.4018/IJEIS.2019070105>
- Pereira dos Santos, R., Lima Werner, C., Barbosa, O., & Alves, C. (2012). Software Ecosystems: Trends and Impacts on Software Engineering. *2012 26th Brazilian Symposium on Software Engineering*, 206-210.
<https://doi.org/10.1109/SBES.2012.24>
- Pereira dos Santos, R., & Lima Werner, C. M. (2012). ReuseECOS: An Approach to Support Global Software

Development through Software Ecosystems. *2012 IEEE Seventh International Conference on Global Software Engineering Workshops*, 60-65. <https://doi.org/10.1109/ICGSEW.2012.16>

Pereira dos Santos, R., & Werner, C. (2011). Treating business dimension in software ecosystems. *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, 197-201. <https://doi.org/10.1145/2077489.2077526>

Raatikainen, M., Kettunen, E., Salonen, A., Komssi, M., Mikkonen, T., & Lehtonen, T. (2021). State of the Practice in Application Programming Interfaces (APIs): A Case Study. En *Software Architecture* (1.^a ed., pp. 191-206). Springer Nature. https://doi.org/10.1007/978-3-030-86044-8_14

Runeson, P., Höst, M., Rainer, A., & Regnell, B. (2012). *Case Study Research in Software Engineering*. Wiley. <https://doi.org/10.1002/9781118181034>

Tan, W., Fan, Y., Ghoneim, A., Hossain, M. A., & Dustdar, S. (2016). From the Service-Oriented Architecture to the Web API Economy. *IEEE Internet Computing*, 20(4), 64-68. <https://doi.org/10.1109/MIC.2016.74>

ANEXOS

Anexo 1. Dimensión Infraestructura

Herramienta/ Servicio/ Recurso	RapidAPI	APILayer	Nubentos	Apilanding
Herramientas para soportar el ciclo de vida.	RapidAPI Client RapidAPI for Teams	API provider Postman – OpenApi – Alojamiento en la Nube (varias opciones)	Espacio de trabajo. Catálogo de APIs. Acceso a Usuarios y configuración de planes de precios.	API-File API-DB
Herramientas para consumir las APIs	RapidAPI Hub RapidAPI for Teams RapidAPI Testing	Live Demo (prueba de APIs) y la función Run Code Catálogo	10 SDK (integración de APIs) - Catálogo.	API-Center API-DOC API-Testing API-Cart
Soporte Técnico	Formulario para reportar problemas	Klaviyo	Chat con experto – centro de ayuda.	Generador de Tickets
Recursos de Documentación	Guías, tutoriales, blogs, Get Started, videos, webinars, ebooks, hojas de datos, reportes y white papers.	Descripción y ejemplos de cada API.	Blog – guías – ebooks – videos – reserva de demo.	Documentación, códigos fuentes de ejemplo (para más de 10 lenguajes), un tutorial completo y documentación técnica, comercial, descriptiva y de casos de éxito por cada API
Monitoreo	RapidAPI Testing (monitoreo global) RapidAPI Hub: Desempeño de APIs (popularidad, latencia promedio, nivel de servicio)	Desempeño de APIs; APIs en uso, límite de solicitudes, por mes/día, solicitudes restantes mes/día Datadog (servicio de monitorización de servidores, bases de datos, herramientas y servicios en la nube)	Desempeño de APIs, nivel de consumo por usuario, preferencias de uso. Desempeño de la plataforma, % de actividad del Gateway, Marketplace y Publisher (30 días).	Desempeño del Gateway: cantidad de llamados recibidos periódicamente dentro de la API-Cloud, visualización de todos los consumos, consumo por tipo de API y fechas de actividad e indicadores (consumos y créditos disponibles)
Otros	RapidAPI Enterprise Hub	Hubspot (CRM) Stripe (pagos) ReferralCandy		

Anexo 2. Dimensión normas

Políticas.	RapidAPI	APILayer	Nubentos	Apilanding
Privacidad de Confidencialidad/datos	X	X	X	X
Seguridad de datos				X
Términos y Condiciones de Uso (Servicios/Plataforma)	X	X	X	X
Derechos de Autor	X			
Política de Cookies	X	X	X	X
Política de Precios			X	

Anexo 3. Dimensión Normas. Modelo de Monetización (consumidores)

Consumidores	RapidAPI	APILayer	Nubentos	Apilanding
Producto / servicios	Productos (RapidAPI Hub, RapidAPI Client, etc.).	Plataforma (soporte técnico, tipos de uso (comercial/personal), soporte de mail, y otras funcionalidades) APIs (Cantidad de solicitudes diarias y mensuales, cantidad de endpoints)	Plataforma Consumo de APIs	Fijo: Panel Administrador, Soporte Cloud, Alta Disponibilidad, Variable: cantidad de llamadas diarias, cantidad de APIs incluidas, registro con una cantidad de créditos
Planes estructurados	Planes gratuitos, freemium o pagos (con cuotas y límites). Suscripción mensual o planes de pago por uso	Planes (free, starter, pro, enterprise y custom)	Plan de Usuario + Plan de Consumo (Free, Freemium, Paid)	Planes (Bronze, Silver, Gold).
Planes flexibles	Planes privados disponibles solo por invitación	Planes personalizados		

Anexo 4. Dimensión Normas. Modelo de Monetización (proveedores)

Proveedores	RapidAPI	APILayer	Nubentos	Apilanding
Gratuito	Publicación de APIs	Publicación de APIs	Publicación de APIs	Publicación de APIs
Cobro	Comisión de transacción sobre los pagos a efectuar al proveedor.	Alojamiento en la Nube.		Comisión de venta sobre lo contratado por los consumidores de APIs.

Anexo 5. Contexto

Objetivo	RapidAPI	APILayer	Nubentos	Apilanding
Establecer Comunicación	Facebook, Twitter Linkedin, YouTube	Facebook, Twitter Linkedin, Instagrams Referral Candy Hotjar - Klaviyo	Facebook, Twitter Linkedin. YouTube	Facebook, Twitter y Linkedin
Extensión / mejora de la Infraestructura	GitHub, PagerDuty, Slack y Twilio.	Postman, GitHub, Servicios de Nube (5 empresas), Docker, Google Analytics, Datadog		
Soporte a normas	DocuSign (contratos)	Hubspot (CRM), Stripe (pagos)		