

El Portal de Integración de IPaaS

Propuesta de comparación aplicando el proceso analítico jerárquico

Sandra Casas¹, Diana Cruz², Graciela Vidal¹, and Marcela Constanzo¹

Instituto de Tecnología Aplicada, Universidad Nacional de la Patagonia Austral,
Argentina

¹ {sicasas,gvidal,mconstanzo}@uarg.unpa.edu.ar
² dcruz@uart.unpa.edu.ar

Resumen. Una plataforma de integración de servicios (IPaaS) es un conjunto de servicios en la nube que permite a los usuarios crear, administrar y gobernar flujos de integración que conectan una amplia gama de aplicaciones o fuentes de datos sin instalar ni administrar ningún hardware o middleware. En los últimos años han emergido una gran cantidad de IPaaS, la selección de las mismas se plantea como un problema para los potenciales usuarios/clientes. Este trabajo propone el análisis y comparación de IPaaS considerando los recursos y servicios disponibles u ofrecidos al desarrollador en la plataforma, lo que denominamos Portal de Integración. Para este análisis se aplicó el método Proceso Analítico Jerárquico (AHP). Siguiendo este enfoque, se ha caracterizado el Portal de Integración, en cinco dimensiones y 29 atributos; y se aplicó a las IPaaS, Zapier, MuleSoft y Celigo.

Keywords: plataforma de integración de servicio · portal del desarrollador · API web · servicios en la nube.

1 Introducción

Las Plataformas de Integración como Servicio (IPaaS) se consideran una solución a los diversos problemas de integración de aplicaciones, al tiempo que a los clientes les evitan los esfuerzos y costos derivados del mantenimiento de las infraestructuras, actualizaciones, y licencias. Además de ofrecer una solución unificada para el intercambio de datos, que resuelve de manera transparente y automática problemas de heterogeneidad de datos, formatos, protocolos, etc. Se ajustan perfectamente a procesos de desarrollo ágil y low-code, lo que las vuelven muy atractivas.

El crecimiento de las IPaaS, en el contexto de la industria API, es notable. El reporte de estado de mercado de API Landscape [1], en relación a las herramientas API y las empresas que impulsan ecosistemas digitales globales, ubica a las IPaaS entre las cinco subcategorías de la industria API, e informa de un financiamiento acumulado hacia fines del 2021 de \$14.7 B, entre 149 ecosistemas habilitados.

Según [2] una IPaaS es un conjunto de servicios en la nube que permite a los usuarios crear, administrar y gobernar flujos de integración que conectan una amplia gama de aplicaciones o fuentes de datos sin instalar ni administrar ningún hardware o middleware.

En general, un portal para desarrolladores debe proporcionar al desarrollador toda la información necesaria para el desarrollo de una aplicación. De [3] hace referencia a un portal para desarrolladores en el contexto de la gestión de APIs Web, que es un caso de uso común de un portal para desarrolladores. Este debe proporcionar documentación y acceso a las APIs e información de análisis de API. Aunque no se ha hallado una descripción o caracterización del portal de integración de IPaaS, consideramos que debe proporcionar diversos recursos y servicios que permiten integrar las aplicaciones haciendo uso de la plataforma. Algunas características básicas de IPaaS se acercan a esta definición, disponer de una interfaz de desarrollo fácil de usar y un amplio conjunto de conectores pre-construidos y plantillas de integración, brindar las herramientas necesarias y los recursos dedicados para un desarrollo rápido de los flujos de integración que conectan diferentes aplicaciones y fuentes de datos [4].

Dado que en los últimos años han emergido una gran cantidad de IPaaS, la selección de las mismas se plantea como problema para los potenciales usuarios/clientes. Es por ello, que existen varios enfoques para la evaluación o comparación de estas plataformas [5–7], entre otros, pero ninguno está enfocado específicamente en el Portal de Integración.

Este trabajo, propone el análisis y comparación de IPaaS considerando las facilidades para la integración de aplicaciones y flujos de datos, disponibles u ofrecidas a los usuarios de la plataforma. Es decir, aquellos recursos, servicios y herramientas que deben formar parte de la plataforma para que el desarrollador la use. Para este análisis se aplicó el método denominado Proceso Analítico Jerárquico (AHP) [8, 9], en tanto que el portal de integración de IPaaS se compone de múltiples elementos, cuyas características y propósitos específicos son diferentes, a la vez que su importancia también difiere. AHP ha sido aplicado ampliamente en la solución de una gran variedad de problemas. Es un método matemático creado para evaluar alternativas cuando se tienen en consideración varios criterios y está basado en el principio que la experiencia y el conocimiento de los actores son tan importantes como los datos utilizados en el proceso. Aplicamos el modelo generado a tres IPaaS que consideramos maduras y consolidadas en el mercado, Zapier, MuleSoft y Celigo.

Los aportes de este estudio son, (1) la identificación y caracterización del Portal de Integración en el contexto de IPaaS, en cinco dimensiones y 29 atributos; (2) la definición de un enfoque para el análisis y comparación de dichas plataformas; y (3) la aplicación de este enfoque a las tres IPaaS mencionadas.

A continuación, en la Sección 2 se presenta el método AHP, en la Sección 3 se describe la metodología aplicada en este estudio, en la Sección 4 se detalla la definición de atributos y criterios de la jerarquía, cuya cuantificación se enuncia en la Sección 5, en la Sección 6 se presentan los resultados de la aplicación del

método AHP y en la Sección 7 se explicita los trabajos relacionados. Finalmente, la Sección 8 expone la discusión y las conclusiones.

2 Proceso Analítico Jerárquico (AHP)

El Proceso Analítico Jerárquico (*Analytic Hierarchy Process*, AHP) es un método propuesto por Thomas L. Saaty, como una respuesta a problemas concretos de toma de decisiones en el Departamento de Defensa de los Estados Unidos [10]. Este método selecciona alternativas en función de una serie de criterios o variables, normalmente jerarquizados, los cuales suelen entrar en conflicto. El objetivo final se encuentra en nivel más alto, y los criterios y subcriterios en niveles inferiores.

Una vez definida la estructura jerárquica, se comparan los criterios de cada grupo del mismo nivel jerárquico y la comparación directa por pares de las alternativas respecto a los criterios del nivel inferior. Para ello se utilizan matrices de comparación pareadas usando una Escala Fundamental (Tabla 1). La clave del método es utilizar esta escala de comparación por pares.

Tabla 1. Escala fundamental de comparación por pares [8]

Valor	Definición
1	Igual importancia
3	Importancia moderada
5	Importancia grande
7	Importancia muy grande
9	Importancia extrema
2, 4, 6 y 8	Valores intermedios entre los anteriores, cuando es necesario matizar

Tanto la comparación de las diferentes alternativas respecto al criterio del nivel inferior de la estructura jerárquica como la comparación de los diferentes criterios de un mismo nivel jerárquico, dan lugar a una matriz cuadrada denominada matriz de decisión. Esta matriz cumple con las propiedades de *reciprocidad* (si $a_{ij} = x$, entonces $a_{ji} = 1/x$), *homogeneidad* (si i y j son igualmente importantes, $a_{ij} = a_{ji} = 1$, y además, $a_{ii} = 1$ para todo i), y *consistencia*. La consistencia se obtiene mediante el *índice de consistencia* (*Consistency Index*, CI) donde λ_{max} es el máximo autovalor y n es la dimensión de la matriz de decisión:

$$CI = \frac{\lambda_{max} - n}{n - 1} \quad (1)$$

Un índice igual a cero significa que la consistencia es completa. Una vez obtenido CI, se obtiene la proporción de consistencia (*Consistency Ratio*, CR) siendo aceptado siempre que no supere el 5% para una matriz de tamaño 3, el 9% para una matriz de tamaño 4 y el 10% para una matriz de tamaño 5 o mayor. Donde *RI* es el *índice aleatorio*, que indica la consistencia de una matriz aleatoria (Tabla 2):

$$CR = \frac{CI}{RI} \quad (2)$$

Tabla 2. Índice aleatorio RI

Tamaño de la matriz	2	3	4	5	6	7	8	9	10
Índice aleatorio	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

Una vez verificada la consistencia, se obtienen los pesos, que representan la importancia relativa de cada criterio o las prioridades de las diferentes alternativas respecto a un determinado criterio. AHP utiliza el método de los autovalores, donde se debe resolver la siguiente ecuación:

$$A \cdot w = \lambda_{max} \cdot w \quad (3)$$

donde A representa la matriz de comparación, w el autovector o vector de preferencia, y λ_{max} el autovalor.

3 Metodología

Este estudio se realizó en dos etapas, la primera consistió en un conjunto de actividades necesarias para la aplicación de AHP, y la segunda etapa siguió el método AHP. A continuación, se describen para cada etapa, las actividades realizadas.

Etapla Definición de la Estructura Jerarquía (Criterios, Atributos e IPaaS): (1) Se procedió a la definición de criterios y atributos. Los portales para integrar aplicaciones, ofrecen una diversidad de servicios y recursos que tienen diversos propósitos, usos y funciones. Un paso necesario es la determinación de criterios o dimensiones que se consideran claves. Luego por cada criterio, se determinan los atributos (componentes) del mismo. Para realizar esta categorización se usó la información de diversas fuentes como [3, 11] y nuestra propia experiencia como desarrolladoras de software. (2) A continuación, se seleccionaron los IPaaS que participarían de la comparación. Se estableció como criterio de inclusión que sean plataformas reconocidas en el mercado y maduras. Con los elementos previos se elaboró el diagrama de la jerarquía. (3) Cuantificación de los IPaaS por criterios. Primero se realizó una indagación en las plataformas de las IPaaS seleccionadas, lo cual implicó, lectura de la documentación, inspección y participación de las comunidades y recursos externos recomendados desde la plataforma, se realizaron consultas a los mails y cuentas oficiales de las redes sociales, se efectuaron pruebas simples de los asistentes, herramientas de soportes, búsquedas, etc. A partir de estas experiencias, se determinaron los atributos disponibles por cada criterio en cada plataforma. Luego, se procedió a valorizar los atributos disponibles de cada criterio y la sumatoria de los mismos se asignó al criterio. Se aplicó un esquema similar al usado en [12] y representado genéricamente en la Tabla 3.

Etapla Aplicación de AHP a las IPaaS: se procedió al cálculo matemático descrito en la Sección 2.

Tabla 3. Esquema de valoración de criterios

Criterios	Atributos	Disponible	Cuantificación
<i>Criterio</i> ₁	<i>Atributo</i> ₁	Si/No	v_1 (Si = 10 / No = 1)
	<i>Atributo</i> ₂	Si/No	v_2 (Si = 10 / No = 1)

	<i>Atributo</i> _{<i>n</i>}	Si/No	v_n (Si = 10 / No = 1)
<i>Valor del criterio</i> ₁			$\sum v_i$
...
<i>Criterio</i> _{<i>n</i>}	<i>Atributo</i> ₁	Si/No	v_1 (Si = 10 / No = 1)

<i>Valor del criterio</i> _{<i>n</i>}			$\sum v_i$

4 Definición de la jerarquía

Para la definición de los criterios y atributos consideramos las descripciones de portales de desarrollador de [3, 11] y nuestra propia experiencia como desarrolladoras de software. Luego se derivó en cinco criterios, y por cada uno se identificaron los atributos que se consideraron más característicos. A continuación, se exponen los criterios y atributos.

Catálogo y búsqueda (#1). Son componentes cuya función es permitir a los desarrolladores encontrar las aplicaciones, APIs o integraciones pre hechas (conectores). Los catálogos (a veces denominados directorios, marketplace o repositorios) están organizados por categorías. La operación de búsqueda permite hallar APIs, Apps o conectores, según un término que se considera clave. Los atributos son, búsqueda, catálogo de API, catálogo de APP, catálogo de conectores.

Documentación (#2). Se considera todo contenido textual o multimedia cuyo propósito es ayudar y facilitar al desarrollador a realizar integraciones en primer lugar, y luego a usar los demás servicios que proporciona la plataforma. En esta categoría de componentes, la comunicación del contenido es unidireccional, ya que no hay interacción con el destinatario de la información. Los atributos de esta categoría son, tutoriales y guías, videos, ejemplos de código, cursos, libros, especificaciones (OpenAPI u otra), artículos y documentación externa (GitHub u otro) que se recomienda desde la plataforma IPaaS.

App de Soporte (#3). Son las herramientas y artefactos a disposición del desarrollador para generar y construir nuevas integraciones. Los atributos que corresponden a este criterio son, kits de desarrollo de software (SDK), editores en línea (plantillas y formularios), interfaces de línea de comandos (CLI) o consolas, herramientas de prueba, herramientas para gestión de versiones y generadores de tickets.

Información de Uso de IPaaS (#4). Corresponde a la información que debe estar disponible en el portal, para un uso adecuado del IPaaS y que excluye a la documentación. Los atributos son, los acuerdo a nivel de servicio (SLD), modelo de monetarización (precios, tarifas y/o planes), políticas del uso de la

6 Sandra Casas, Diana Cruz, Graciela Vidal, and Marcela Constanzo

plataforma (términos de uso, política de privacidad, términos de servicio, política de cookies) y analíticas.

Comunidad (#5). Se considera todo recurso o servicio que tiene el mismo propósito que la documentación, pero permite alguna forma de interacción con los usuarios o entre ellos, y por lo tanto es dinámico. Los atributos de esta dimensión son, los foros o grupos de discusión, wikis o blogs, formularios de ayuda (mailing), chatbots o asistentes virtuales en línea, cuentas en redes sociales, espacios de discusión en plataformas o repositorios externos específicos (Stack Overflow u otro) que se difunde desde la plataforma IPaaS y presentaciones en línea (webinar y meeting).

Los portales de integración analizados corresponden a las IPaaS Zapier, MuleSoft y Celigo. Actualmente, detentan de popularidad y madurez en el mercado, con diferencias de prestaciones y capacidades, marcan tendencias, en lo que refiere al panorama de plataformas de integración.

En la Fig. 1 se presenta la jerarquía, de acuerdo a los criterios previamente definidos.

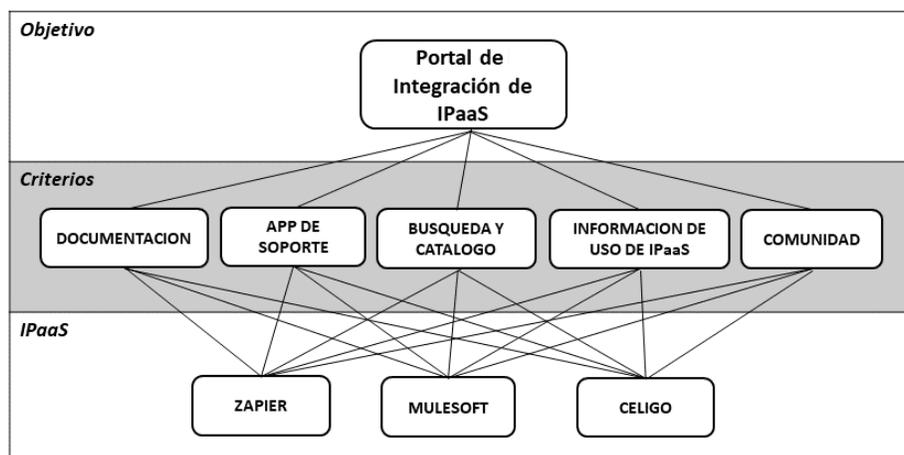


Fig. 1. Diagrama de jerarquía: Objetivo, Criterios y Alternativas

5 Cuantificación de Atributos y Criterios

A continuación, se presenta el análisis resumido de los atributos de los cinco criterios por cada IPaaS, y luego en la Tabla 4, se especifica la cuantificación de criterios y atributos según la Tabla 3.

Zapier. La documentación está compuesta por una extensa guía que detalla los pasos para realizar integraciones y cómo usar toda la plataforma, incluye imágenes, videos cortos y ejemplos de código. Una gran cantidad de artículos cortos

en los que los lectores pueden hacer comentarios. Los cursos (videos largos) se disponen en la sección Universidad. Desde la plataforma se enlaza a GitHub en el cual se encuentra un tutorial muy completo. Las integraciones pueden realizarse con dos herramientas, un CLI para crear integraciones en código JavaScript desde el entorno de desarrollo local, y un generador visual en línea que para crear integraciones mediante formularios y plantillas. Las distintas partes de una integración se pueden probar de varias formas. Mantiene un historial de las pruebas y el desarrollador puede monitorear la traza de las mismas paso a paso. Zapier permite manejar distintas versiones de una integración. Se puede estar desarrollando la nueva versión, mientras se ejecuta la anterior. Al lanzar una nueva versión se pueden migrar los usuarios y ofrece un procedimiento de aviso (deprecar) a los usuarios antes de desplegar la nueva versión. El directorio de Zapier ofrece más de 4000 App para integrar (Google, Discord, Trello, etc.). Están clasificadas por diferentes categorías (función/dominio/empresa y otros criterios). Permite la búsqueda por la denominación de la app. Las integraciones realizadas en Zapier también están disponibles en el catálogo. Zapier dispone de todas las políticas de plataforma de manera bien detallada. Ofrece un modelo de monetización basado en tarifas, con cinco planes (free, starter, profesional, equipo y compañía), que se definen en torno a cantidad de tareas y servicios ofrecidos. Los SLD se definen como restricciones operativas de Zapier y la integración, y se establecen en términos de tiempo de ejecución (segundos) y carga útil (filas, campos y mb) en tres etapas del desarrollo de la integración. Al crear una integración desde uno de los editores se accede a algunas analíticas (uso de zaps, triggers, acciones, habilitados, en pausa). Después de hacer pública una aplicación, se podrán ver la cantidad de usuarios activos mensual y trimestralmente. La comunidad esta organiza en foros (por temas y subtemas) y grupos cerrados coordinados por expertos. Desde la plataforma se difunden las cuentas oficiales en cuatro redes sociales y un grupo de discusión en Stack Overflow. Se dispone de un formulario para enviar consultas (mail).

MuleSoft. La plataforma es un producto de Salesforce. La documentación se compone de guías detalladas con imágenes, modelos y esquemas de las diferentes funcionalidades que ofrece al desarrollador para realizar integraciones y automatizar procesos de negocio, con un soporte de tutoriales en videos cortos y whitepapers. Además, MuleSoft mantiene una gran variedad de recursos complementarios como e-books, webinars, blogs, artículos e infografías. Las integraciones pueden realizarse mediante una herramienta Anypoint CLI, un editor en línea y un IDE AnyPoint Studio Desktop para Eclipse. Además, MuleSoft ofrece el Mule SDK para Java y XML para extender al Mule 4 Runtime y así facilitar la creación, el desacople y consistencia de módulos. MUnit es un framework de pruebas de aplicaciones Mule que se encuentra integrado completamente en Anypoint Studio y permite construir pruebas automatizadas para probar integraciones y APIs, y generar informes de cobertura. También provee la versión plugin Maven para ejecutar pruebas MUnit como parte de entornos de integración continua. Después de crear una API en AnyPoint Platform, se puede configurar los detalles de la versión y publicarla en Anypoint Exchange. Las distintas versiones

que administra MuleSoft se conocen como activos de Exchange. Se pueden agregar, eliminar y desaprobar versiones de una API. Ante incidentes operativos, Mulesoft brinda un proceso de generación de tickets de soporte para gestionar casos y elegir la gravedad correcta antes de crear un nuevo ticket. Anypoint Exchange es un catálogo seleccionado de activos reusables. Hasta el momento, el catálogo se agrupa en tres categorías generales, tales como, activos proporcionados por Mulesoft y sus socios, activos para salud, y activos para comunicaciones. El catálogo está clasificado en 13 tipos de activos (APIs, políticas, fragmentos de especificaciones de API, grupos de API, templates, integraciones, ejemplos, conectores, y otros cinco tipos más), que ayuda a especificar la búsqueda. En este catálogo se puede publicar, compartir, descubrir y reusar activos para facilitar la colaboración, aumentar la productividad y promover estándares. El catálogo mantiene la información de las versiones de los activos publicados. Salesforce establece políticas relacionadas a términos de uso, cookies, seguridad, privacidad y cumplimiento para el producto MuleSoft. A su vez, en Mulesoft ofrece los Gateway Flex y Mule para configurar políticas en las API de los usuarios, tales como controlar la autenticación, el acceso, el consumo asignado y el SLA (Service Level Access). Mulesoft permite configurar las analíticas que se necesitan para proporcionar información sobre cómo se utilizan las API y crear paneles de administración personalizados. La comunidad se organiza en foros por temas y subtemas, grupos de discusión se mantiene un blog de comunicación y anuncios, y ofrece grupos de reuniones por regiones que organizan eventos con expertos. Además, se ofrece a la comunidad comunicarse mediante cuentas oficiales en cuatro redes sociales. El soporte de ayuda ofrece un asistente solo para usuarios registrados y subscriptos a un producto pago. Es posible contactar por medio de un chatboot y via telefónica.

Celigo. La plataforma permite integrar Apps y automatizar procesos de negocios. La documentación se encuentra disponible en diferentes formatos. Sobre el editor integrator.io se accede a un asistente en línea que permite realizar búsquedas y explorar diferentes recursos como tours sobre la plataforma, tutoriales, ejemplos y videos. Otra forma de acceder a la documentación es el centro de ayuda, por medio de búsquedas, acceso a links rápidos y a artículos relacionados a integrator.io, automatización de procesos de negocios, capacitaciones y grupos de integración. Al explorar la plataforma es posible realizar preguntas a un agente de soporte por medio de un chat. A través de Celigo University y Webinars ofrece más de 100 cursos y seminarios destinados a usuarios y socios de distintos niveles. También provee dos e-books en tres idiomas. El editor integrator.io permite configurar flujos e integraciones personalizadas o utilizar plantillas de integraciones. Si la aplicación requerida no está disponible en la plataforma puede realizarse utilizando adaptadores universales (HTTP o REST API). Mientras se lleva a cabo la configuración de integraciones y flujos se pueden ejecutar pruebas por medio del módulo Error Manangement 2.0 para la gestión y documentación de errores. De cada ejecución se registra flujo, estado, duración, errores, estado, clasificación, también proporciona sugerencias para resolver errores automáticamente. Celigo presenta un catálogo con más de 200 Apps de

integración y de procesos de negocios (Adobe sign, Amazon, Ebay, Wish, Zoom, etc.), las cuales son accedidas mediante una búsqueda o su exploración. Las apps se encuentran agrupadas por categoría y por tipo de proceso. Dentro de estos dos grupos existen categorías basadas en el área de aplicación y funcionalidad de las Apps (ERP, PIM, IT Operations, Gestión de proyectos, Gastos de reembolso, Gestión de artículos, etc.). Celigo establece políticas relacionadas a términos de uso, cookies, seguridad, privacidad y cumplimiento. Los SLA están incorporados al Acuerdo de Suscripción de Servicios (ASS), tienen una duración trimestral y garantizan disponibilidad de servicios (99,95% Tiempo activo), mantenimiento programados (sábados 8 pm a 2 am) y no programados, cálculo de créditos y procesos de créditos SLA. La plataforma dispone de un modelo de monetización basado en planes (Estándar, Profesional, Premium y Enterprise), y las tarifas, que varían de acuerdo a la cantidad de endpoints, apps de integración, flujos, socios comerciales y gestión de APIs. Además, ofrece una versión gratuita por 30 días. La opción Analytics de integrator.io muestra información útil para detectar patrones o tendencias para luego tomar decisiones. La comunidad de Celigo ocupa un lugar importante, desde el centro de ayuda se accede a diferentes contenidos destinados a los usuarios. Permite realizar búsquedas, acceder a posts de miembros de Celigo, seguir a los autores, compartir posts en diferentes redes sociales y realizar comentarios. La opción enviar Ticket posibilita enviar consultas sobre tópicos y apps, indicar la urgencia y preferencia de horarios para recibir la respuesta.

6 Aplicación de AHP

La Tabla 5 es la cuantificación final de los criterios de cada IPaaS, obtenida de la Tabla 4.

En (4) se presentan dos matrices y un vector. La primer matriz corresponde a la comparación de criterios aplicando la escala fundamental de comparación por pares, Tabla 1, (la primer fila y columna corresponde al criterio #1, la segunda fila y columna al criterio #2, y así, sucesivamente). Aquí hemos considerado que las Apps de Soporte y la Documentación tienen la mayor importancia en el Portal del Integración, seguido de la Búsqueda y Catálogo, y la Información de Uso del IPaaS, finalmente Comunidad. La segunda matriz es la normalización de la primera, y posteriormente, el vector de pesos o preferencias, que indica la importancia relativa de los criterios. Donde refleja que los criterios #2 y #3, tienen igual relevancia pero son los principales entre los cinco. En esta instancia, el índice de consistencia es válido, con un valor calculado de 0.09.

$$\begin{pmatrix} \mathbf{1.00} & 0.20 & 0.20 & 1.00 & 5.00 \\ 5.00 & \mathbf{1.00} & 1.00 & 5.00 & 7.00 \\ 5.00 & 1.00 & \mathbf{1.00} & 5.00 & 7.00 \\ 1.00 & 0.20 & 0.20 & \mathbf{1.00} & 5.00 \\ 0.20 & 0.14 & 0.14 & 0.20 & \mathbf{1.00} \end{pmatrix} \begin{pmatrix} 0.08 & 0.08 & 0.08 & 0.08 & 0.20 \\ 0.41 & 0.39 & 0.39 & 0.41 & 0.28 \\ 0.41 & 0.39 & 0.39 & 0.41 & 0.28 \\ 0.08 & 0.08 & 0.08 & 0.08 & 0.20 \\ 0.02 & 0.06 & 0.06 & 0.02 & 0.04 \end{pmatrix} \begin{pmatrix} 0.10 \\ 0.38 \\ 0.38 \\ 0.10 \\ 0.04 \end{pmatrix} \quad (4)$$

10 Sandra Casas, Diana Cruz, Graciela Vidal, and Marcela Constanzo

Tabla 4. Cuantificación de criterios y atributos de las IPaaS

Criterio	Atributo	Zapier	MuleSoft	Celigo
Catálogo y Búsqueda	Búsqueda	10	10	10
	Catálogo de APIs	1	10	1
	Catálogo de App	10	1	10
	Catálogo de Integraciones	10	10	10
		31	31	31
Documentación	Tutorial / Guías	10	10	10
	Ejemplos	10	10	10
	Videos	10	10	10
	Cursos	10	10	1
	Documentación externa	10	1	10
	Libros	1	10	10
	Especificación	1	10	1
	Artículos	10	10	10
		62	71	62
App de soporte	Editor en línea	10	10	10
	SDK	1	10	10
	CLI	10	10	10
	Generador de pruebas	10	10	10
	Gestión de versiones	10	10	10
	Generador de tickets	1	10	10
		42	60	60
Comunidad	Foros y/o grupos	10	10	10
	Blogs y/o wikis	10	10	10
	Redes sociales	10	10	10
	Mail	10	1	10
	Chatbot / Asistente en línea	1	10	10
	Otros espacios de discusión	10	1	10
	Webinar / meeting	10	10	10
	61	52	70	
Información de Uso de IPaaS	Políticas y Términos de uso y servicio	10	10	10
	SLD	10	10	10
	Análíticas	10	10	10
	Monetización	10	10	1
	40	40	31	

Tabla 5. Cuantificación final de criterios de las IPaaS

Criterios	Zapier	MuleSoft	Celigo
Búsqueda y Catálogo (#1)	31	31	31
Documentación (#2)	62	71	62
App de Soporte (#3)	42	60	60
Información de Uso de la IPaaS (#4)	70	70	61
Comunidad (#5)	61	52	70

A continuación, se presentan las matrices obtenidas de aplicar el proceso de comparación por pares, por cada criterio, según los valores obtenidos en la Tabla 5, las normalizadas y los vectores de preferencias. En (5) corresponde al criterio Búsqueda y Catálogo (#1), en (6) a Documentación (#2), en (7) a App de Soporte (#3), en (8) a Información de Uso del IPaaS (#4), y en (9) a Comunidad (#5). Los índices de consistencia han resultado, 0, 0.03, 0.6, 0, y 0.7 respectivamente. Debe considerarse que el orden de filas y columnas es, Zapier, MuleSoft y Celigo.

$$\begin{pmatrix} \mathbf{1.00} & 1.00 & 1.00 \\ 1.00 & \mathbf{1.00} & 1.00 \\ 1.00 & 1.00 & \mathbf{1.00} \end{pmatrix} \begin{pmatrix} 0.33 & 0.33 & 0.33 \\ 0.33 & 0.33 & 0.33 \\ 0.33 & 0.33 & 0.33 \end{pmatrix} \begin{pmatrix} 0.33 \\ 0.33 \\ 0.33 \end{pmatrix} \quad (5)$$

$$\begin{pmatrix} \mathbf{1.00} & 0.33 & 1.00 \\ 3.00 & \mathbf{1.00} & 3.00 \\ 1.00 & 0.33 & \mathbf{1.00} \end{pmatrix} \begin{pmatrix} 0.20 & 0.20 & 0.20 \\ 0.60 & 0.60 & 0.60 \\ 0.20 & 0.20 & 0.20 \end{pmatrix} \begin{pmatrix} 0.20 \\ 0.60 \\ 0.20 \end{pmatrix} \quad (6)$$

$$\begin{pmatrix} \mathbf{1.00} & 0.20 & 0.20 \\ 5.00 & \mathbf{1.00} & 1.00 \\ 5.00 & 1.00 & \mathbf{1.00} \end{pmatrix} \begin{pmatrix} 0.09 & 0.09 & 0.09 \\ 0.45 & 0.45 & 0.45 \\ 0.45 & 0.45 & 0.45 \end{pmatrix} \begin{pmatrix} 0.09 \\ 0.45 \\ 0.45 \end{pmatrix} \quad (7)$$

$$\begin{pmatrix} \mathbf{1.00} & 1.00 & 3.00 \\ 1.00 & \mathbf{1.00} & 3.00 \\ 0.33 & 0.33 & \mathbf{1.00} \end{pmatrix} \begin{pmatrix} 0.43 & 0.43 & 0.43 \\ 0.43 & 0.43 & 0.43 \\ 0.14 & 0.14 & 0.14 \end{pmatrix} \begin{pmatrix} 0.43 \\ 0.43 \\ 0.14 \end{pmatrix} \quad (8)$$

$$\begin{pmatrix} \mathbf{1.00} & 3.00 & 0.33 \\ 0.33 & \mathbf{1.00} & 0.20 \\ 3.00 & 5.00 & \mathbf{1.00} \end{pmatrix} \begin{pmatrix} 0.23 & 0.33 & 0.22 \\ 0.08 & 0.11 & 0.13 \\ 0.69 & 0.56 & 0.65 \end{pmatrix} \begin{pmatrix} 0.26 \\ 0.11 \\ 0.63 \end{pmatrix} \quad (9)$$

Finalmente en (10), se muestran los resultados finales del proceso. La matriz surge de los vectores obtenidos (5), (6), (7), (8), (9), que representan por cada criterio la prioridad lograda en cada alternativa. Dicha matriz y el vector de importancia relativa de los criterios (4), determinan finalmente las prioridades de las tres IPaaS, que se han representado en términos porcentuales. De acuerdo a los valores que se han asignado en cada paso, resulta que el portal de integración de Mulesoft es el más completo.

$$\begin{array}{c} \text{Zapier} \\ \text{MuleSoft} \\ \text{Celigo} \end{array} \begin{matrix} \text{\#1} & \text{\#2} & \text{\#3} & \text{\#4} & \text{\#5} \\ \begin{pmatrix} 0.33 & 0.20 & 0.09 & 0.43 & 0.26 \\ 0.33 & 0.60 & 0.45 & 0.43 & 0.11 \\ 0.33 & 0.20 & 0.45 & 0.14 & 0.63 \end{pmatrix} \end{matrix} \begin{pmatrix} 19.88\% \\ 48.13\% \\ 31.99\% \end{pmatrix} \quad (10)$$

7 Trabajos Relacionados

En [13] analizaron las IPaaS con respecto a la independencia de la plataforma, la usabilidad, la facilidad de programación y la mantenibilidad. Dividieron las propiedades en tres grupos: alcance de la herramienta, capacidades de modelado y características técnicas. Compararon las IPaaS, Camel, MuleSoft, ServiceMix, Spring Integration y BizTalk. Esta propuesta se centra en las propiedades generales, mientras que nuestra propuesta proporciona un método para comparar plataformas de integración con respecto al portal de integración.

En [14] analizaron características que impactan en la productividad, escalabilidad, elasticidad, tiempo de ciclo e incorporaron en la calidad de los equipos de TI, reportando sus experiencias sobre el uso de las plataformas de integración Amazon SQS y Boomi. Comparado con nuestra propuesta, además de analizar diferentes propiedades, el método que aplicaron no fue descripto, mientras que nuestra propuesta presenta la metodología aplicada y por ende es replicable.

En [15] propusieron redes de Petri para simular modelos conceptuales de soluciones de integración diseñados con la plataforma Guaraná [16], como método de evaluación de la calidad. El objetivo de nuestra propuesta es diferente.

En [17] analizaron y evaluaron las plataformas de integración mediante una taxonomía y la aplicaron a las plataformas, Boomi, Informatica, MuleSoft y SAP. Utilizaron dos grupos de criterios: funcionales y no funcionales. Los criterios funcionales fueron procesamiento de ejecución, número de operadores, conectividad, administración y desarrollo. Los criterios no funcionales estudiados fueron: precio, contrato de servicio, confiabilidad, tecnología, seguridad y gestión del servicio. Este trabajo tiene un enfoque de propósito general.

En [5] se presenta un framework para la comparación y evaluación de IPaaS con respecto al rendimiento de los sistemas de tiempo de ejecución. Se compone de diez propiedades de rendimiento organizadas en dos dimensiones, procesamiento de mensajes y balance de ejecución. Los autores analizan y comparan nueve IPaaS Open Source. En [12] los mismos autores, presentan una experiencia en la evaluación y comparación de cuatro plataformas de integración de código abierto, en el que el rendimiento era un requisito central para elegir una plataforma de integración. La evaluación se realizó utilizando una adaptación de AHP. La evaluación objetiva se basa en propiedades distribuidas en tres dimensiones, procesamiento de mensajes, detección de puntos críticos y ejecución justa, que componen la metodología de investigación que utilizamos. Las plataformas evaluadas fueron clasificadas para identificar la de mejor desempeño.

En [6], los autores proponen identificar las prácticas de integración y las barreras relacionadas con el diseño con respecto a la adopción de IPaaS. A partir de un estudio cualitativo, que se basó en 10 entrevistas con expertos de diferentes proveedores de software como servicios, informan que los principales factores relacionados con la adopción de IPaaS son la estandarización de los modelos de datos, la facilidad de uso y la variedad de conectores proporcionados, y los problemas relacionados con la privacidad, seguridad y transparencia de los datos.

Desde una perspectiva comercial y orientado al sector de la industria, se encuentran varias propuestas de evaluación. G2.com ofrece un ranking de 38 API web Marketplaces, en los que están incluidos diversas IPaaS. Este ranking califica a los productos y proveedores en función de las reseñas recopiladas de la comunidad de usuarios, así como de los datos agregados de fuentes en línea y redes sociales. Aplican un algoritmo a estos datos para calcular las puntuaciones de Satisfacción del cliente y Presencia en el mercado en tiempo real. Ovum (ahora OMDIA) presenta la matriz de decisión de Ovum [7] como una evaluación integral para ayudar a los potenciales clientes (arquitectos, directores,

gerentes, líderes) a seleccionar la iPaaS que mejor se adapte a sus requisitos específicos. Realizaron cuestionarios y reuniones informativas con los proveedores, en relación a diversos tópicos que se organizaron en tres dimensiones: evaluación de tecnología, evaluación de la ejecución y evaluación de impacto en el mercado. Gartner Research, publica un ranking de iPaaS [18], el cual está basado en las revisiones de usuarios en torno a cuatro ejes, evaluación y contratación, integración y despliegue, servicio y soporte, y capacidades del producto.

8 Discusión y Conclusiones

Este trabajo presentó un enfoque para analizar y comparar los recursos y servicios ofrecidos para desarrollar integraciones de las iPaaS. Se aplicó el proceso analítico jerárquico, para ello se establecieron cinco criterios o dimensiones, Búsqueda y Catálogo (#1), Documentación (#2), App de Soporte (#3), Información de Uso del iPaaS (#4), y Comunidad (#5), los cuales se descompusieron en 29 atributos. A partir de nuestra propia experiencia, consideramos que las dimensiones Documentación y Aplicaciones de Soporte resultan las más importantes cuando un desarrollador decide usar una iPaaS, dado que son los componentes básicos que se requieren para usar la iPaaS. Se analizaron las iPaaS, Zapier, MuleSoft y Celigo, siendo, Mulesoft la que obtuvo la prioridad.

Otros autores han estudiado la evaluación y comparación de iPaaS, pero se centran en propiedades generales o aspectos más técnicos como el rendimiento, el desempeño en ejecución, la seguridad, etc.

Una limitación que observamos en nuestra propuesta, está referida a la cuantificación de los criterios y atributos. Aplicamos el mismo enfoque que [12], cuyo objetivo era comparar iPaaS en términos del rendimiento. Diversos atributos, fueron calificados con el valor 1 o 10, para representar la ausencia o presencia de propiedades tales como la ejecución de políticas, identificación de patrones y el disponer de una política para controlar la tasa de mensajes entrantes. Sin embargo, en nuestro caso, al solo considerar la presencia o ausencia del atributo, dos iPaaS pueden obtener el mismo valor, cuando incluso haya diferencias entre ambos. Por ejemplo, todas las iPaaS cuentan con artículos, pero en el caso de MuleSoft se trata de “white papers”, mientras que en Zapier los artículos son cortos y estructurados. De igual manera, respecto del atributo “analíticas”, existen diferencias en cuanto a la cantidad y variedad de datos que se ofrecen al desarrollador. En este sentido, cuando dos atributos obtienen la ponderación igual a 10, no se reflejan estas posibles discrepancias.

El trabajo futuro está precisamente dirigido a resolver estas situaciones. Para ello, se analizarán indicadores y/o métricas, objetivos o subjetivos, factibles de ser aplicados para cada atributo.

Referencias

1. Apiday, Platformable: API Landscape of the Market 2022, <https://apilandscape.apiscene.io/>, last accessed 2022/05/13.

14 Sandra Casas, Diana Cruz, Graciela Vidal, and Marcela Constanzo

2. Serrano, N., Hernantes, J., Gallardo, G.: Service-oriented architecture and legacy systems. *IEEE Softw.* 31, 15–19 (2014). <https://doi.org/10.1109/MS.2014.125>.
3. De, B.: API Management. In: *API Management*. pp. 15–28. Apress, Berkeley, CA (2017).
4. Marian, M.: iPaaS: Different Ways of Thinking. *Procedia Econ. Financ.* 3, 1093–1098 (2012). [https://doi.org/10.1016/S2212-5671\(12\)00279-1](https://doi.org/10.1016/S2212-5671(12)00279-1).
5. Freire, D.L., Frantz, R.Z., Roos-Frantz, F., Sawicki, S.: Survey on the run-time systems of enterprise application integration platforms focusing on performance. *Softw. Pract. Exp.* 49, 341–360 (2019). <https://doi.org/10.1002/SPE.2670>.
6. Neifer, T., Lawo, D., Bossauer, P., Gadatsch, A.: Decoding iPaaS: Investigation of user requirements for integration platforms as a service. *Proc. 18th Int. Conf. e-Business, ICE-B 2021*. 47–55 (2021).
7. Sharma, S.: Ovum Decision Matrix highlights the growing importance of iPaaS and API platforms in hybrid integration. Ovum Consulting, London, UK (2017).
8. Saaty, T.L.: The analytical hierarchy process, planning, priority. *Prior. Setting. Resour. Alloc. MacGraw-Hill, New York Int. B. Co.* 287 (1980).
9. Saaty, T.: Decision making with the Analytic Hierarchy Process. *Int. J. Serv. Sci.* 1, 83–98 (2008). <https://doi.org/10.1504/IJSSCI.2008.017590>.
10. Piqueras Yepes, V.: Proceso Analítico Jerárquico (Analytic Hierarchy Process, AHP), <https://victoryepes.blogs.upv.es/2018/11/27/proceso-analitico-jerarquico-ahp/>, last accessed 2022/05/13.
11. Careem, M.: Building an API Strategy Using an Enterprise API Marketplace, <https://wso2.com/whitepapers/building-an-api-strategy-using-an-enterprise-api-marketplace/>, last accessed 2022/05/13.
12. Freire, D.L., Frantz, R.Z., Roos-Frantz, F.: Ranking enterprise application integration platforms from a performance perspective: An experience report. *Softw. Pract. Exp.* 49, 921–941 (2019). <https://doi.org/10.1002/SPE.2679>.
13. Corchuelo, R., Frantz, R., González, J.: Una Comparación De ESBs Desde La Perspectiva de La Integración De Aplicaciones. In: *Jornadas de Ingeniería del Software y Bases de Datos (JISBD)*. pp. 403–408 (2008).
14. More, V., Bartere, M.: Enterprise Integration Using Boomi Tool. *Int. J. Adv. Inf. Sci. Technol.* 11, 18–22 (2013).
15. Roos-Frantz, F., M. Binelo, R. Z. Frantz, S. Sawicki, and V. B. Fernandes. 2015. “Using Petri Nets to Enable the Simulation of Application Integration Solutions Conceptual Models.” *Conference on Enterprise Information Systems(ICEIS)*, 87–96.
16. Frantz, R. Z., R. Corchuelo, and C. Molina-Jimenez. 2012. “A Proposal to Detect Errors in Enterprise Application Integration Solutions.” *Journal of Systems and Software* 85 (3): 480–497. doi:10.1016/j.jss.2011.10.048.
17. Ebert, N., Weber, K.: Integration Platform as a Service in der Praxis: Eine Bestandsaufnahme. In: *Multikonferenz Wirtschaftsinformatik (MKWI)*. pp. 1675–1685 (2016).
18. Guttridge, K., Pezzini, M., Golluscio, E., Thoo, E., Iijima, K., Wilcox, M.: Magic quadrant for enterprise integration platform as a service. Gartner Inc., Stanford, CT (2017).