UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR

POSGRADOS

MAESTRÍA EN

SOFTWARE CON MENCIÓN EN DISEÑO DE ARQUITECTURA DE SISTEMAS

RPC-SO-34-NO.778-2021

OPCIÓN DE TITULACIÓN:

PROYECTO DE TITULACIÓN CON COMPONENTES DE INVESTIGACIÓN APLICADA Y/O DE DESARROLLO

TEMA:

DISEÑO DE UNA ARQUITECTURA BASADA EN API MANAGER PARA EL PAGO DE NÓMINA EN LÍNEA PARA EL SEGMENTO EMPRESAS

AUTOR(ES)

FAUSTO FERNANDO AMAGUAÑA GALARZA

DIRECTOR:

PATSY MALENA PRIETO VÉLEZ

QUITO – ECUADOR

2025

Autor(es):

Fausto Fernando Amaguaña Galarza
Ingeniero en Sistemas.
Candidato a Magíster en Software con Mención en Diseño de Arquitectura de Sistemas por la Universidad Politécnica Salesiana – Sede Quito.
famaguana@est.ups.edu.ec

Dirigido por:

Patsy Malena Prieto Vélez
Ingeniera en Sistemas Informáticos y de Computación.
Magíster en Gestión Informática Empresarial.
Máster Universitario en Ciencias y Tecnologías de la Información.
pprieto@ups.edu.ec

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra para fines comerciales, sin contar con autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual. Se permite la libre difusión de este texto con fines académicos investigativos por cualquier medio, con la debida notificación a los autores.

DERECHOS RESERVADOS
2025© Universidad Politécnica Salesiana.
QUITO– ECUADOR – SUDAMÉRICA
Fausto Fernando Amaguaña Galarza
DISEÑO DE UNA ARQUITECTURA BASADA EN API MANAGER PARA EL PAGO DE NÓMINA EN LÍNEA PARA EL SEGMENTO EMPRESAS.

DEDICATORIA

Dedico este trabajo

A Dios, quien ha guiado cada paso de mi vida, tanto en lo personal como en lo profesional, dándome la fuerza necesaria para alcanzar un nuevo peldaño en mi formación académica.

A mis padres, Fausto Amaguaña y Norma Galarza, quienes han sido el pilar fundamental en mi trayectoria educativa y un ejemplo de trabajo, esfuerzo, constancia y dedicación.

Finalmente, a mis familiares, amigos y compañeros, cuyo apoyo moral ha sido invaluable en los momentos que más lo necesité.

AGRADECIMIENTO

Quiero expresar mi agradecimiento a Dios, quien me da brindado su fortaleza y paciencia en los momentos difíciles.

A la Universidad Politécnica Salesiana, quien me ha formado como ingeniero y ahora en magíster, brindándome las herramientas necesarias para ser un profesional de calidad.

A mi tutora Patsy Prieto, por ser un guía en todo este camino, brindándome retroalimentación y orientación para poder culminar mi tesis.

A mis padres Fausto y Norma, por los valores que me han inculcado a lo largo de mi vida.

Y finalmente a mis familiares, amigos y compañeros, quienes me ayudaron si le requería ya sea con una palabra de aliento o con un consejo.

TABLA DE CONTENIDO

ÍNDICE DE FIGURAS 7
ÍNDICE DE TABLAS 8
Resumen 10
ABSTRACT 11
1. INTRODUCCIÓN 12
1.1 ANTECEDENTES 12
1.2 DETERMINACIÓN DE PROBLEMA 14
1.2.1 DESCRIPCIÓN DEL PROBLEMA 14
1.2.2 FORMULACIÓN DEL PROBLEMA 15
1.3 JUSTIFICACIÓN DEL PROBLEMA 16
1.4 DELIMITACIÓN DEL PROBLEMA 16
1.5 OBJETIVOS 17
1.5.1 OBJETIVO GENERAL 17
1.5.2 OBJETIVOS ESPECÍFICOS 17
MARCO TEÓRICO REFERENCIAL 18
2.1 TOGAF 18
2.2 ENTIDAD FINANCIERA 20
2.3 EMPRESA 20
2.3.1 PYMES 21
2.3.2 PYMES EN ECUADOR 21
2.4 NÓMINA 22
2.4.1 GESTIÓN DE NÓMINA: 22
2.4.2 ADMINISTRACIÓN DE NÓMINA AUTOMATIZADO 23
2.5 PAGO EN LÍNEA 24
2.6 BIAN (BANKING INDUSTRY ARCHITECTURE NETWORK) 24
2.7 SOFTWARE FINANCIERO 26
2.7.1 DESARROLLO DE SOFTWARE 27
Figura 4: Metodología Ágil 28
2.8 ARQUITECTURA API 29
2.8.1 COMPONENTES DE LA ARQUITECTURA API 30
2.8.2 PASOS PARA EL DISEÑO DE LA ARQUITECTURA API 30
2.8.3 CAPAS DE ARQUITECTURA API 31
2.9 MICROSERVICIOS 31

2.10 DISEÑO DE ARQUITECTURA DE MICROSERVICIOS. 32
2.11 DISEÑO DE LA ARQUITECTURA EMPRESARIAL 34
DESARROLLO DEL PROYECTO 38
3.1 MÉTODO DE DESARROLLO DE LA ARQUITECTURA (ADM) DE TOGAF 38
3.1.1 FASE A: VISIÓN DE LA ARQUITECTURA 38
3.1.2 FASE B: ARQUITECTURA EMPRESARIAL 49
3.1.3 FASE C: ARQUITECTURAS DE SISTEMAS DE INFORMACIÓN (APLICACIÓN) 52
3.1.4 FASE D: ARQUITECTURA TECNOLÓGICA 57
3.1.5 FASES E & F: OPORTUNIDADES, SOLUCIONES Y PLANIFICACIÓN DE LA MIGRACIÓN. 59
RESULTADOS Y DISCUSIÓN 61
4.1 IMPLEMENTACIÓN BIAN 61
4.2 IMPLEMENTACIÓN ARQUITECTURA DE APIS PARA EL PAGO DE NÓMINA EN LÍNEA 66
4.2.1 DIAGRAMA DE CONTEXTO 66
4.2.2 DIAGRAMA DE CONTENEDOR 68
4.2.3 DIAGRAMA DE COMPONENTES 69
CONCLUSIONES 72
GLOSARIO 74
REFERENCIAS 76

ÍNDICE DE FIGURAS

Figura 1: Descripción general de la arquitectura Portal v.12.0 25
Figura 2: Diagramas de modelos de objetos de negocio BIAN versión 12.0 26
Figura 3: Metodología Cascada 28
Figura 4: Metodología Ágil 28
Figura 5: ADM El Ciclo del Método de Desarrollo de la Arquitectura. 35
Figura 6: Stakeholder o partes interesadas de este proyecto. 39
Figura 7: Objetivos comerciales asociados con la rentabilidad de la entidad financiera. 42
Figura 8: Principios de arquitectura. 43
Figura 9: Refinamiento, Objetivo Estrategia Afiliación 44
Figura 10: Estrategia a largo plazo 47
Figura 11: Concepto de la solución**.** 49
Figura 12: Estructura de la organización. 50
Figura 13: Descomposición de la organización 50
Figura 14: Flujo de valor. 51
Figura 15: Cooperación de aplicaciones (Línea base). 52
Figura 16: Cooperación de aplicaciones (Objetivo). 53
Figura 17: Uso de la Aplicación Api Mánager Pago Nómina. 55
Figura 18: Uso de la Aplicación Api Mánager Pago Nómina Objetivo. 55
Figura 19: Estructura de información muestra los principales objetos comerciales. 57
Figura 20: Infraestructura (objetivo). 58
Figura 21: Comunicación Servicios Recibir Orden Pago Nómina. 59
Figura 22: Vista de migración. 60
Figura 23: BIAN Service Domain Landscape version 12.0. 62
Figura 24: BIAN Service Landscape V12.0 Value Chain View 63
Figura 25: Descripción General del Servicio de Dominio de Nómina Corporativa 64
Figura 26: Diagrama BOM de Servicios de Nómina Corporativa 64
Figura 27: Diagrama de registro de control de servicios de nómina corporativa 65
Figura 28: BIAN-Servicios de Nómina Corporativa. 66
Figura 29: Diagrama de Contexto 67
Figura 30: Diagrama de Contenedor. 69
Figura 31: Diagrama de Componentes**.** 71

ÍNDICE DE TABLAS

Tabla 1: **Componentes de la arquitectura API.** 30
Tabla 2**: Capas de arquitectura API.** 31
Tabla 3: **Método de Desarrollo de la Arquitectura (ADM) de TOGAF**..... 38
Tabla 4: **BIAN Service Landscape V12.0.**..... 63

DISEÑO DE UNA ARQUITECTURA BASADA EN API MANAGER PARA EL PAGO DE NÓMINA EN LÍNEA PARA EL SEGMENTO EMPRESAS.

Autor:

Fausto Fernando Amaguaña Galarza

Resumen

El presente trabajo identificó un problema en el servicio de pago de nómina que ofrece una entidad financiera a las empresas afiliadas. La necesidad de mejorar este servicio se derivó de la insatisfacción de los usuarios y la falta de eficiencia en los procesos actuales, lo que afectó la continuidad del negocio y los objetivos de la institución. Para abordar esta problemática, se propuso un diseño que permitió la transición de una arquitectura monolítica a una arquitectura más ágil y eficiente.

El objetivo de la investigación fue diseñar una solución para optimizar el servicio de pago de nómina mediante la implementación de una arquitectura de software que cumpliera con los estándares de la industria financiera. Para ello, se empleó el framework de Esquema de Arquitectura del Open Group (TOGAF) para el análisis de requerimientos, lo que facilitó la identificación de una arquitectura transitoria que sirviera de puente hacia una arquitectura objetivo. Además, se incorporó el uso de Interfaces de Programación de Aplicaciones (API) estandarizadas bajo principios REST, siguiendo el marco de referencia de la Red de Arquitectura de la Industria Bancaria (BIAN), para garantizar la interoperabilidad y eficiencia en la gestión de servicios, para gestionar estas Apis se establece el uso de Api Manager.

En conclusión, los principales hallazgos indicaron que la adopción de un enfoque ágil, el estándar TOGAF, combinado con el uso de APIs estandarizadas, fue crucial para llegar al planteamiento de una propuesta efectiva para la mejora del servicio de pago de nómina. La utilización de herramientas de modelado como Archimate y el estándar C4 durante las fases del ciclo TOGAF también contribuyó al desarrollo de una solución integral y efectiva.

Palabras clave:

Servicio, Pago de nómina, Insatisfacción, Eficiencia, Arquitectura, API mánager, BIAN, TOGAF

ABSTRACT

The present work identified an issue within the payroll payment service provided by a financial entity to its affiliated companies. The need to enhance this service arose from user dissatisfaction and the inefficiency of current processes, which adversely impacted business continuity and the institution's objectives. To address this issue, a design was proposed that facilitated the transition from monolithic architecture to a more agile and efficient architecture.

The objective of the research was to design a solution that optimized the payroll payment service through the implementation of software architecture that adhered to industry standards within the financial sector. To achieve this, The Open Group Architecture Framework (TOGAF) was employed for requirements analysis, which facilitated the identification of a transitional architecture that would serve as a bridge to the target architecture. Additionally, the use of standardized APIs based on REST principles was incorporated, following the Banking Industry Architecture Network (BIAN) reference framework, to ensure interoperability and efficiency in service management. To manage these APIs, the use of an API Manager is established.

In conclusion, the main findings indicated that the adoption of an agile approach, the TOGAF standard, combined with the use of standardized APIs, was crucial to reaching the approach of an effective proposal for the improvement of the payroll payment service. The use of modeling tools such as Archimate and the C4 standard during the TOGAF cycle phases also contributed to the development of a comprehensive and effective solution.

Keywords:

Service, Payroll, Dissatisfaction, Efficiency, Architecture, APIs (Application Programming Interfaces), BIAN, TOGAF.

1. INTRODUCCIÓN

En el competitivo panorama empresarial en el que se encuentran las empresas hoy en día, exige a las organizaciones ser no sólo eficientes, sino también innovadoras. Sin embargo, muchas empresas enfrentan dificultades en la gestión de sus sistemas contables, en particular en lo que respecta al proceso de pago de nómina. El problema clave que se identificó es la falta de agilidad y eficiencia en el servicio de pago de nómina proporcionado por una entidad financiera a sus afiliados, lo que puede resultar en pérdidas económicas significativas y afectar negativamente la satisfacción del cliente.

Se ha diseñado una arquitectura basada en API Manager que optimiza el proceso de pago de nómina en línea. Esta solución tiene como objetivo mejorar la eficacia del servicio y garantizar la estabilidad operativa al reducir los tiempos de transacción y minimizar los errores asociados con la gestión manual.

1.1 ANTECEDENTES

El pago en línea es un servicio que ofrecen las instituciones financieras a sus clientes bajo un modelo de negocio Business To Customer (B2C), es decir, de la entidad financiera a sus clientes o consumidores. Cientos de transacciones se procesan a diario a través de los canales físico y digitales disponibles en el Ecuador.

A diferencia de cómo se manejan los pagos para empresas, que se realizan de forma masiva, el pago a los empleados de la empresa afiliada puede incluir a personas no bancarizados. La entidad financiera provee una plataforma web que requiere aprobación manual y la generación de archivos para procesar pagos como nómina, proveedores o terceros. Las empresas generan estos archivos internamente y los cargan en la aplicación de la entidad. Si hay errores, el proceso debe repetirse, lo que puede retrasar las transacciones durante días.

El proceso de aprobación implica roles y perfiles adecuados para garantizar la seguridad e integridad de los datos. La seguridad es un pilar fundamental en el diseño del sistema para prevenir ataques (Nosrati & Bidgoli, 2015). La piratería de cuentas bancarias ha causado pérdidas millonarias en todo el mundo, lo que resalta la importancia de la seguridad en los sistemas bancarios y su impacto en las instituciones financieras y sus clientes.

La banca global también forma parte del cambio tecnológico, modernizando sus métodos, tecnologías y formas de hacer negocios para adaptarse a la percepción del cliente sobre la nueva banca digital. Estos cambios responden a las nuevas necesidades surgidas en la sociedad como consecuencia de la pandemia (Begum & Raaj, 2021).

La tecnología en la banca moderna ha evolucionado significativamente, pasando de la externalización de funciones administrativas como Contabilidad y Nómina (outsourcing), a enfoques electrónicos centralizados y automatizados. Las rápidas transformaciones en el sector financiero están impulsadas por el crecimiento de los mercados económicos y globales. Esto ha dado lugar a conceptos como Open Banking, OpenAPI y BIAN, que responden a la necesidad de desarrollar APIs para permitir el acceso a datos por parte de terceros a través de plataformas abiertas, convirtiéndose en una tendencia clave en la industria financiera.

Open Banking busca fomentar la innovación y el uso de nuevas tecnologías, permitiendo que las entidades financieras compartan datos de forma segura mediante API. OpenAPI es un formato estándar que define la estructura y las reglas de las API REST, facilitando su desarrollo y uso. Por su parte, BIAN es una organización sin fines de lucro que desarrolla especificaciones de OpenAPI para servicios financieros. Para gestionar y administrar eficientemente estas APIs, surge el API Manager, una herramienta que permite crear, publicar, monitorear y gestionar conexiones, facilitando la integración con diferentes componentes de software.

Se plantea un reto tecnológico también a las empresas que desean ser parte de esta transformación digital, ya que la banca hace esfuerzos por poner a su disposición sus servicios, canales de atención físicos y digitales (Pathak & Joshi, 2009). La expansión de Internet elevó la banca a un nuevo nivel y ha planteado enormes oportunidades de transacciones conjuntas.

En este caso de negocio de pago de nómina, las empresas son las propietarias de la información, la cual es considerada como datos personales de sus empleados. Por esta razón, el Reglamento General de Protección de Datos (GDPR) establece los principios que rigen el tratamiento de dicha información, lo que requiere que el intercambio de los datos se realice de manera estandarizada. Sólo los datos necesarios para la orquestación del pago serán compartidos con la entidad financiera.

Otro reto que presentan los grandes volúmenes de información es la visualización de los datos ya procesados en tiempo real. Dentro de las entidades bancarias se están utilizando herramientas de inteligencia empresarial para la gestión del análisis y el monitoreo de datos (Coppolino, 2015).

Las APIs son herramientas fundamentales para escalar la integración B2B (business to business), es decir, "de empresa a empresa". Son APIs expuestas a socios y clientes, lo que reduce la barrera para integrar procesos comerciales e intercambiar datos financieros, en este caso, entre la entidad financiera y las empresas.

1.2 DETERMINACIÓN DE PROBLEMA

1.2.1 DESCRIPCIÓN DEL PROBLEMA

En el marco de la Revolución Industrial 4.0, las tecnologías emergentes como la inteligencia artificial, el aprendizaje automático y las soluciones en la nube están transformando los sistemas financieros a nivel global. Sin embargo, en Ecuador, las instituciones financieras tradicionales enfrentan serios desafíos para adaptarse a esta nueva era de transformación digital. La integración ineficaz de estas tecnologías y las crecientes preocupaciones relacionadas con la ciberseguridad, así

como las estrictas regulaciones de organismos de control y las normativas sobre protección de datos personales, limitan significativamente su capacidad para ser competitivas y ofrecer servicios innovadores que satisfagan las demandas del mercado.

En particular, los procesos de pago de nómina en las instituciones financieras presentan ineficiencias que impactan negativamente tanto en el segmento empresarial como en la administración de empleados, proveedores, mantenimiento y la recepción de alícuotas. Estas deficiencias no solo ralentizan el crecimiento, sino que también comprometen la estabilidad financiera del sector, dificultando su adaptación a los estándares de la industria financiera a nivel internacional.

Este escenario plantea la necesidad de investigar estrategias que permitan a las instituciones financieras en Ecuador implementar tecnologías avanzadas orientadas a la nube para mejorar los sistemas de pago de nóminas. Dichas estrategias deben estar alineadas con la seguridad, el cumplimiento normativo y las necesidades de las empresas en proceso de transformación digital, garantizando un equilibrio entre innovación y estabilidad financiera.

1.2.2 FORMULACIÓN DEL PROBLEMA

Los segmentos empresariales, corporativo y Pyme, que incluyen empresas grandes, medianas y pequeñas, requieren productos financieros como el pago de nómina y actualmente utilizan una plataforma proporcionada por la entidad financiera. Estas empresas deben seguir un flujo de afiliación extendido a través de una aplicación web con arquitectura monolítica, asistidas por un analista de negocios que las guía durante el proceso de onboarding, el cual abarca todos los aspectos técnicos y legales. Al finalizar este proceso, que puede llevar un tiempo considerable a pesar de ser asistido, las empresas deben capacitarse en el uso de la plataforma.

La aplicación web cuenta con todos los flujos y políticas de seguridad necesarias para garantizar transacciones seguras. Por esta razón, el proceso requiere de actores con responsabilidades específicas, como aprobadores, administradores y

controladores, que deben gestionar manualmente la plataforma de la entidad financiera. Este proceso conlleva un tiempo de gestión y posterior procesamiento de las órdenes de pago de nómina. Sin embargo, esto representa una oportunidad para la entidad financiera de mejorar la experiencia de sus clientes actuales y atraer nuevos, para aumentar su transaccionalidad y generar mayores ingresos.

1.3 JUSTIFICACIÓN DEL PROBLEMA

La entidad financiera cuenta con un proceso definido y una plataforma web con infraestructura limitada a la solución existente. Esta plataforma permite a sus actuales clientes del servicio de pago de nómina gestionar el proceso mediante la carga de órdenes de pago en formatos de archivos planos, con el flujo de aprobación y las medidas de seguridad requeridas. Sin embargo, para alcanzar sus metas y continuar con su crecimiento, es fundamental identificar nuevas oportunidades de negocio. Para ello, utilizamos TOGAF, que nos ayuda a identificar estas oportunidades a través de las fases que describe este framework, enfocándonos en una arquitectura adecuada que permita a la entidad financiera aumentar su transaccionalidad y garantizar la continuidad del negocio, beneficiando así a más empresas.

1.4 DELIMITACIÓN DEL PROBLEMA

El desarrollo de este análisis y diseño está orientado a una institución financiera que se encuentre en proceso de transformación digital y busque aumentar su transaccionalidad en el pago de nómina. A continuación, se presenta una lista de aspectos que están fuera del alcance de este documento al no contar con información de alguna institución financiera en específico. El marco de trabajo TOGAF guía hacia la consecución de los objetivos. Para ello, se asume que ciertos procesos y flujos ya forman parte de la entidad financiera y están incluidos en su arquitectura actual. En consecuencia, los procesos complementarios no se abordan en la sección de arquitectura de este proyecto.

- No se contemplan los procesos internos de las empresas para la gestión de nómina.
- No se incluyen los procesos internos de la entidad financiera relacionados con la obtención de datos de los clientes para el desarrollo de este trabajo, tales como costos del servicio, uso del canal, y contratos de confidencialidad entre las partes, etc.
- No se abordan los procesos internos de reportes conciliatorios o reportes para organismos de control de la empresa o entidad financiera.
- No se consideran los procesos tributarios de las empresas o de la entidad financiera.
- No se contempla el cobro de comisiones a las empresas por la utilización del servicio, ni ningún mecanismo de parametrización relacionado.
- Cualquier desarrollo o modificación de los aplicativos existentes en la entidad financiera o empresa, que no se describa explícitamente en este documento, queda fuera de este alcance.
- Cualquier aspecto que no esté explícitamente detallado en este documento también está fuera del alcance del proyecto.

1.5 OBJETIVOS

1.5.1 OBJETIVO GENERAL

Analizar y diseñar una arquitectura para una entidad financiera que gestione el pago de nómina en línea para el segmento de empresas, utilizando API Manager.

1.5.2 OBJETIVOS ESPECÍFICOS

- Identificar los requisitos y limitaciones técnicas y de negocio que surgen al exponer un API para el pago de nómina de los clientes del segmento empresarial de una institución financiera.
- Diseñar el ciclo de vida que proporcione un mecanismo de comunicación entre la empresa y la entidad financiera para el API de pago de nómina en línea.

- Determinar una estructura de datos para el intercambio de información de una API genérica que permita a las empresas afiliadas integrarse al servicio financiero de pago de nómina.

MARCO TEÓRICO REFERENCIAL

En este capítulo es importante realizar una descripción de los conceptos que van a intervenir en el diseño de una arquitectura basada en Apis para el pago de nómina en línea para el segmento empresas, tanto de negocio como técnicas para la comprensión de este proyecto.

2.1 TOGAF

TOGAF, es un framework estructurado orientado a resolver, mejorar y optimizar las funcionalidades e integraciones de sistemas o procesos, con el objetivo de cumplir las metas del negocio y de sus partes interesadas (Sandoval, 2017; Josey, 2013).
El framework TOGAF diagnostica el estado inicial, evalúa alternativas probadas o referenciales, plantea un escenario alcanzable y planifica su implementación. Para ello, emplea el Método de Desarrollo de Arquitectura (ADM), que propone cuatro fases principales. Estas fases se utilizarán de acuerdo con el alcance de este trabajo:

Fase A: Visión de la Arquitectura
Es la fase inicial, donde se identifican los stakeholders (partes interesadas), se analizan sus expectativas y se definen los requerimientos del negocio que se desean alcanzar.

El resultado de la fase es:

- Identificar los Stakeholder
- Identificar los objetivos comerciales asociados a la rentabilidad.
- Principios de Arquitectura.
- Visión de la arquitectura.

- Refinamiento de los principios de negocio, metas de negocio, identificar los impulsores de negocio.
- Estrategia a Largo Plazo.
- Concepto de la solución.

Fase B: Arquitectura de Negocio

En esta fase se determina la estructura de la entidad financiera y sus objetivos de negocio. Además, se identifican los procesos, roles y funciones clave.

El resultado de esta fase es obtener:

- Vista de organización.
- Descomposición de la organización (anidada).
- Flujo de valor.

Fase C: Arquitectura de Sistemas de Información

Durante esta fase se establecen los sistemas de información de la entidad financiera, así como los procesos que administran su diseño, procesamiento, integraciones y transformación.

El resultado de la fase es obtener:

- Vista de cooperación de aplicaciones.
- Vista de cooperación de aplicaciones (objetivo).
- Vista de estructura de información que muestra los principales objetos comerciales del pago de nómina.
- Vista de uso de la aplicación.
- Vista de uso de la aplicación (objetivo).

Fase D: Arquitectura Tecnológica

Finalmente, esta fase define la infraestructura tecnológica, incluyendo hardware, protocolos, sistemas de soporte, sistemas de información y tecnologías de comunicación.

2.2 ENTIDAD FINANCIERA

Una entidad financiera en Ecuador es una institución que ofrece servicios y productos financieros, facilitando el intercambio de bienes y servicios y gestionando los medios de pago. Estas entidades buscan el crecimiento económico de sus segmentos: personas y empresas.

Las entidades financieras en Ecuador pueden ser públicas o privadas, y entre ellas se encuentran los bancos, las cooperativas de ahorro & crédito, y las mutualistas. El sistema financiero del país está controlado por la Superintendencia de Bancos.

2.3 EMPRESA

Una empresa es una organización dedicada a una actividad económica que ofrece bienes o servicios con fines de lucro. En Ecuador, las empresas se dividen según su tamaño, de acuerdo con las normativas de la Comunidad Andina de Naciones (CAN), y estas son: microempresas, pequeñas y medianas empresas (pymes) y grandes empresas.

Para Mero (2018), “una empresa es una entidad que, mediante la organización de elementos humanos, materiales, técnicos y financieros, proporciona bienes o servicios a cambio de un precio que le permite la reposición de recursos y la consecución de determinados objetivos”.

Microempresas: Son empresas de tamaño pequeño, en las que trabajan un número reducido de empleados. En varios casos, las microempresas son emprendimientos o negocios familiares, y el número de colaboradores varía entre 1 y 9 personas.

Grandes empresas: Son corporaciones de gran poder comercial que, en algunos casos, operan a nivel internacional. Tienen un mayor alcance económico e influencia en el mercado, y el número de colaboradores supera las 200 personas. Asimismo, se consideran grandes empresas aquellas corporaciones que han

sobrepasado ciertos límites, frecuentemente asociados con la cantidad de trabajadores que poseen y la cantidad de dinero que generan (CEUPE, s.f.).

2.3.1 PYMES

Las PYMES son las pequeñas y medianas empresas, definidas dependiendo del número de trabajadores. Las pequeñas empresas cuentan con aproximadamente de 10 a 49 empleados, mientras que las medianas tienen entre 50 y 199. Las PYMES contribuyen a la economía de un país generando empleo, desarrollo e innovación; sin embargo, enfrentan el desafío de competir con las grandes empresas (Yungrid & Carlos, 2012).

En Ecuador, las PYMES han ido en aumento. Dado este crecimiento, es fundamental que cuenten con un sistema óptimo para el pago de nómina, ya que se están desarrollando gradualmente con el fin último de competir mano a mano con las grandes empresas (Primicias, s.f.).

2.3.2 PYMES EN ECUADOR

Como se ha mencionado anteriormente, las PYMES han ido aumentando en el país, impulsando la economía de Ecuador. Estas empresas tienen las siguientes fortalezas:

- Contribución a la economía: Las PYMES representan más del 90% de las unidades productivas, generan el 60% del empleo, participan en el 50% de la producción y generan casi el 99% de los servicios que un ecuatoriano utiliza en un día. Ejemplos de estos servicios incluyen tiendas, almuerzos, copias, cyber cafés, transporte público, entre otros.
- Capacidad de adaptación: Al contar con un número reducido de trabajadores, las PYMES tienen estructuras organizacionales que se adaptan más rápidamente a los cambios en la economía.
- Innovación: Las PYMES ecuatorianas han sido responsables de numerosos emprendimientos y han logrado exportar sus productos al mercado internacional, gracias a la flexibilidad que ofrece ser emprendedor. La

mayor parte de las innovaciones tecnológicas ha sido desarrollada por PYMES.

- Distribución de ingresos: En las PYMES del país, los salarios de los cargos gerenciales son más cercanos a los de los demás trabajadores de la empresa. Esto contrasta con las grandes empresas, donde un gerente general puede ganar cientos de veces más que un empleado promedio. Esta particularidad de las PYMES contribuye a una mejor redistribución de la riqueza en la economía.

A pesar de estas fortalezas, las PYMES en el país enfrentan desventajas, como la falta de suficiente liquidez, que puede conducir al cierre definitivo de la empresa en caso de desastres naturales o problemas financieros. Además, la falta de experiencia o conocimiento en el ámbito empresarial puede ser un obstáculo. Por esta razón, en Ecuador se ha desarrollado la Ley del Emprendimiento e Innovación, que establece un marco normativo para incentivar y fomentar la creación, desarrollo, crecimiento y expansión de proyectos de emprendimiento a nivel nacional (Primicias, s.f.).

2.4 NÓMINA

Es un documento que se registra los salarios y días trabajados de los empleados de una empresa, además de los beneficios y deducciones, proporcionando el total a pagar a cada día trabajado en un tiempo determinado. Para Benavides una nómina cumple con las siguientes funciones:

Contiene la estrategia de remuneración de una empresa, que puede incluir bonificaciones, licencias y deducciones. También proporciona un desglose detallado del salario. Se utiliza para pagar sueldos o salarios, para recopilar insumos relacionados con la nómica (Laura & Mariela, 2022).

2.4.1 GESTIÓN DE NÓMINA:

El informe financiero de los empleados es un documento en el que se registran, gestionan y organizan los datos financieros del personal, tales como el salario, los

beneficios de ley (Instituto Ecuatoriano de Seguridad Social) y las horas extras, con el objetivo de realizar el pago de nómina.

Según Benavides (2022), "mantener una nómina es extremadamente importante para las actividades cotidianas y también para el cumplimiento de los requisitos legales". En la mayoría de los casos, las empresas utilizan un software, generalmente semi automatizado, para mantener una base de datos de nómina (Laura & Mariela, 2022).

2.4.2 ADMINISTRACIÓN DE NÓMINA AUTOMATIZADO

“Si un trabajador cuenta con la seguridad de que su nómina de pagos está siendo gestionada de manera adecuada, se otorga un mejor nivel de confianza que se traduce en compromiso hacia su lugar de trabajo” (Gualli, 2019).

Esto evidencia que todos los trabajadores deben gozar de sus derechos, los cuales implican los estatutos, la seguridad y las políticas internas que garantizan la obtención de sus remuneraciones, ya sean mensuales o quincenales, de manera equitativa y justa.

Es importante considerar el uso de tecnologías que permitan contar con procesos transparentes y, al mismo tiempo, seguros para todas las partes involucradas.

Por lo tanto, es necesario hacer hincapié en la definición de los roles de pago. Según Bonilla (2019), “el rol de pago se realiza al final del mes y consta del pago quincenal o anticipo; esto busca optimizar el control de los ingresos y egresos, así como los descuentos”. De esta manera, se podrán sintetizar estos procesos, que permitirán brindar resultados confiables.

Es importante señalar que cada organización o empresa diseña sus roles de pago de forma diferente e independiente, dependiendo del tipo de negocio. El propósito de automatizar los cálculos que se procesan en el pago de sueldos mensuales para los trabajadores es mejorar la eficiencia en el uso del tiempo y de los recursos. Por lo

tanto, es esencial enfatizar la confianza, la confiabilidad y la capacidad de proporcionar seguridad tanto a los empleadores como a los empleados.

2.5 PAGO EN LÍNEA

Es un servicio bancario, es una transacción que se realiza de forma electrónica, estos pagos se hacen mediante tarjetas de crédito o débito, dinero electrónico, pasarelas de pagos, entre otros.

Para Hernández (2013), "Los medios de pagos electrónicos se han convertido en una forma de agilizar los procesos económicos modernos, brindando al individuo un mayor número de alternativas a la hora de transferir dinero. La aceptación de estos medios en la sociedad está dada fundamentalmente por el cumplimiento de una serie de características que hacen referencia a aspectos importantes relacionados con los medios de pagos electrónicos"(Evelin & Norma, 2013).

2.6 BIAN (BANKING INDUSTRY ARCHITECTURE NETWORK)

Es una propuesta de arquitectura para estandarizar todos los servicios bancarios, este fue creado en 2008 y sin fines de lucro. (BB Summit, s.f.), esta propuesta tiene como objetivo central apoyar a los negocios y clientes del sector bancario, garantizando altos niveles de rendimiento y seguridad.

El propósito principal es ofrecer al mundo una arquitectura bancaria de excelencia, estableciendo un estándar tecnológico que pueda ser adoptado por instituciones financieras globales. Esta iniciativa tiene como meta proporcionar una hoja de ruta confiable que promueva la innovación constante en el ámbito bancario.

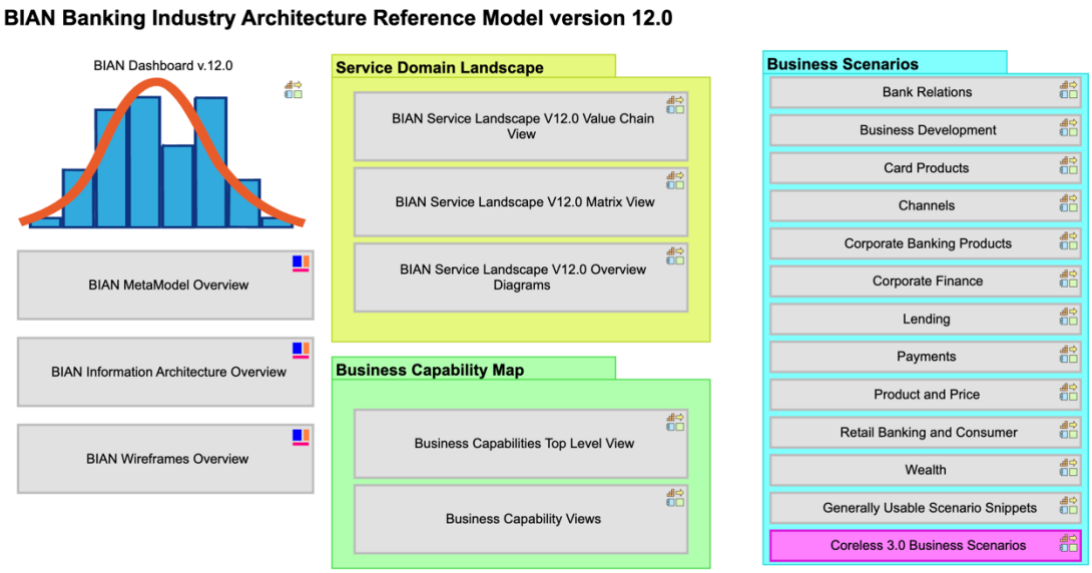
Además, se busca crear un marco de mejores prácticas arquitectónicas en el que los bancos de todo el mundo puedan confiar plenamente. Para lograr esto, se reúnen las mentes más brillantes en el campo de la arquitectura bancaria, fomentando un espacio de colaboración abierta y compartida.

BIAN propone un modelo desacoplado, denominado CORELESS BANKING. Para ello, ha desarrollado artefactos que permiten a la entidad financiera evolucionar en diferentes aspectos de su arquitectura digital. Los aplicables para este caso de negocio son:

- BIAN SERVICES Domains: Es la visión y definición de la organización por DOMINIOS en lugar de procesos; esta es la base de los microservicios.
- BIAN Semantic APIs: Es la nomenclatura estándar para la comunicación en escenarios digitales.
- BIAN BOM Model: Es el modelo semántico de DATOS definido por BIAN e ISO 20022 para escenarios DIGITALES.

En el Portal BIAN visualizamos los artefactos o marcos de referencia de la arquitectura.

Figura 1: Descripción general de la arquitectura Portal v.12.0



Fuente: BIAN(2025)

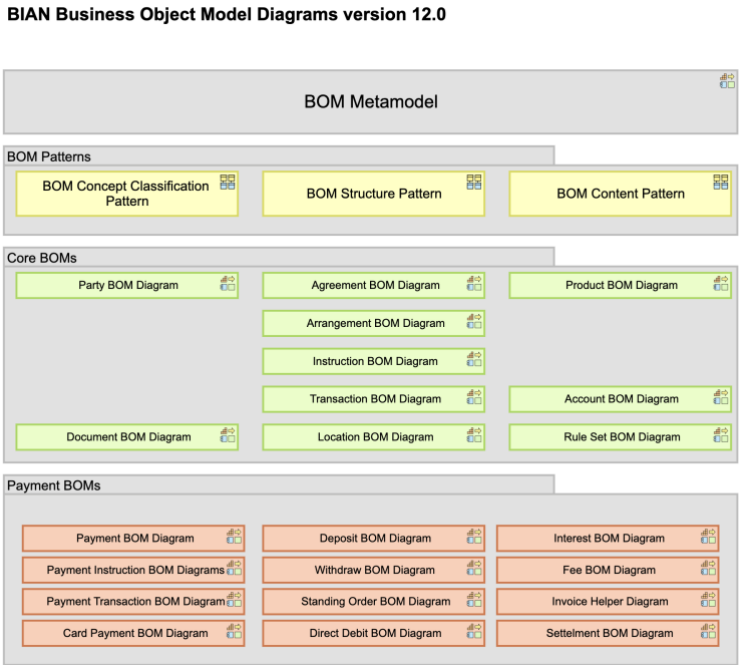
En la descripción general de la arquitectura de información de BIAN, encontramos los Diagramas de Modelos de Objetos de negocio (BOM), que es el modelado de datos abstractos, permite estructurar los datos de manera clara y consistente en los sistemas bancarios propuestos por tres patrones:

El Patrón de Clasificación de Conceptos de BOM: se usa para agrupar las entidades dentro de la entidad de negocio. Por ejemplo, las cuentas corrientes y las cuentas de ahorros, que, debido a sus diferencias, como los sobregiros o límites de transacciones ayudan al monitoreo interno o externo.

Patrón Estructural de BOM: define su composición dentro de una entidad, sus atributos que lo conforman. Por ejemplo, para una Cuenta sus atributos son: titular o dueño, numero, saldo. Esto contribuye a mejorar la interoperabilidad entre los sistemas financieros.

Patrón de Contenido de BOM: Se enfoca a las relaciones de las entidades dentro del modelo, por ejemplo, las relaciones entre el Cliente y la Cuenta Bancaria; un cliente puede tener una o más cuentas, una cuenta puede estar asociado a un o más clientes según el tipo de cuenta.

Figura 2: Diagramas de modelos de objetos de negocio BIAN versión 12.0



Fuente: BIAN(2025)

2.7 SOFTWARE FINANCIERO

Debemos referirnos al software financiero que provee herramientas especializadas para la gestión de las finanzas, tanto de personas naturales como de empresas. Este

software debe ser altamente integrable y de bajo costo, de manera que cubra las necesidades de las empresas que requieren integrarse al sistema financiero. Por ejemplo, el pago de nómina es proporcionado por las entidades financieras, las cuales son controladas por el Estado a través del Banco Central y de la Superintendencia de Bancos.

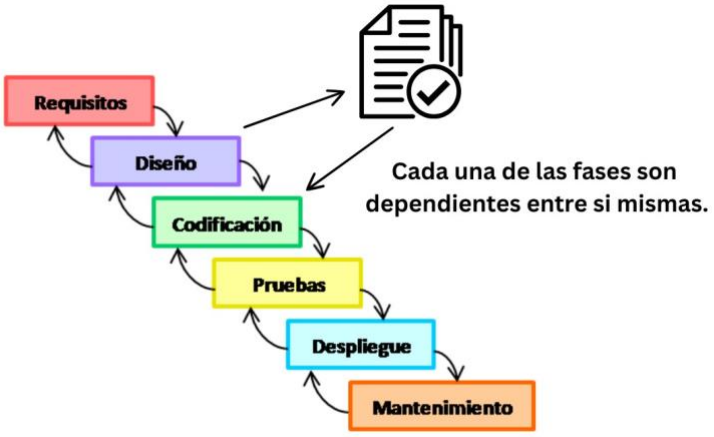
Por otro lado, existen programas de software diseñados para ayudar a las empresas en la gestión de procesos empresariales. Estos sistemas son de alta complejidad en cuanto a su integración y suelen tener un alto costo; un ejemplo de esto son los sistemas de planificación empresarial (ERP).

2.7.1 DESARROLLO DE SOFTWARE

Para el desarrollo, se aplica el Ciclo de Vida del Desarrollo de Software (SDLC), especificando las fases de Planificación, Análisis de Requisitos, Diseño, Implementación, Testeo y Mantenimiento. Para el desarrollo de software existen varias metodologías, siendo las principales la tradicional de cascada y ágil.

La metodología de cascada tiene como característica principal que cada paso depende del anterior, lo que genera un tiempo de espera o Down time entre pasos. Esto permite que los requisitos sean documentados y aprobados, evitando así que sean ambiguos, ya que esta aprobación es un prerrequisito para iniciar el siguiente paso. Sin embargo, si algún paso produce retrasos o desviaciones, la tecnología queda estancada hasta que se genere un nuevo ciclo para realizar cambios, tanto para reducir la deuda técnica como para agregar nuevas funcionalidades. (Figura 3).

Figura 3: Metodología Cascada



Fuente: Propia

Por otro lado, la metodología ágil se caracteriza por su ejecución en ciclos cortos, con entregas orientadas a incrementar el valor al cliente. Es decir, se entregan funcionalidades testeables en producción, aun cuando el producto final requiera de más ciclos. Estos ciclos pueden trabajarse en paralelo o por otros equipos de desarrollo, ya que no deben depender entre sí. Esta metodología está orientada a mejorar la eficiencia y la entrega de valor en tiempos cortos. (Figura 4)

Figura 4: Metodología Ágil

Fuente: Propia

2.8 ARQUITECTURA API

Las API son un protocolo de comunicación entre dos aplicaciones o incluso entre servicios web. Esto permite a un tercero integrarse para hacer uso de las funcionalidades o datos que estas exponen (Ruiz, 2020).

Por lo tanto, gracias a su integración, se mejora el funcionamiento del sector financiero en cuanto al manejo de su base de datos y las aplicaciones que utilizan para atender a sus clientes, ofreciendo mayor fiabilidad y una variedad de servicios. Además, los procesos de definición de las metodologías y los procedimientos que se deben desarrollar son esenciales para evidenciar una buena ejecución de la API. Estos procesos se basan en conjuntos de múltiples componentes y en descripciones de las interacciones lógicas.

Es evidente que las Apis poseen un sinfín de beneficios, como la administración y la creación adecuada de microservicios, teniendo en cuenta las diversas estrategias que se enfocan en impulsar valores. Se hace énfasis en que se pueden exponer las múltiples funcionalidades de los microservicios como productos, donde se crean los potenciales de valores comerciales externos e internos (Bodega, 2019a).

Existen tres tipos de APIs: las APIs privadas, que tienen acceso restringido y solo los miembros de la organización pueden acceder a la información; las APIs de socios, que son abiertas entre los bancos y sus socios en cuanto a intercambiar datos; y las APIs públicas, que comparten información limitada con cualquier persona. Las APIs cuentan con varios métodos de seguridad en relación con la información de las empresas y los clientes; dicha información es cuidadosamente proporcionada por la empresa y evaluada por una entidad regulatoria.

Por lo tanto, gracias a su integración, permite al sector financiero tener un mejor funcionamiento en cuanto al manejo de su base de datos, y las aplicaciones que manejan su cliente, ofreciendo a su cliente mayor fiabilidad y servicios varios. Además, son los procesos de las definiciones de las metodologías y los procedimientos que se deben de desarrollar, para que se evidencie una buena

ejecución de la API. Estas radican de los conjuntos de los múltiples componentes y de las descripciones de las interacciones lógicas.

Evidenciándose que las Apis poseen un sin número de beneficios, como son la administración y la creación adecuada de microservicios, teniendo en cuenta las diversas estrategias que se enfocan e impulsan por los valores, haciendo énfasis que se pueden exponer las múltiples funcionalidades de los microservicios como productos, en donde se crean los potenciales de valores comerciales externos e internos (Bodega, 2019a).

2.8.1 COMPONENTES DE LA ARQUITECTURA API

A continuación, se va a detallar una serie de componentes, los cuales ayudan a la creación de una arquitectura API, según (Valladarez, 2020) manifiesta la siguiente: Tabla 1.

Tabla 1: **Componentes de la arquitectura API.**

Componente	Definición	Función
Puerta de enlace API	Estos componentes se ocupan de las entregas, del detalle específico de las seguridades, de los almacenamientos en caché y orquestaciones.	Ofrecen la capacidad crucial de los equilibrios entre la partición y la carga de los microservicios
Portal API	Son los portales que existen entre los consumidores y los proveedores de API.	Analiza, documenta, es compatible con el marketing, en la web y pueden ser móviles.

Fuente: (Valladarez, 2020)

2.8.2 PASOS PARA EL DISEÑO DE LA ARQUITECTURA API

Es importante tener en cuenta que las APIs son enlaces que se manifiestan entre los usuarios, las tecnologías y la empresa. Considerando que el diseño de esta API debe ser estricto con el valor de los modelos comerciales, es obligatorio seguir las

siguientes consideraciones, que son cruciales y contienen información para la toma de decisiones sobre el correcto diseño de la API en las empresas:

- Alinear el diseño de la API con los propósitos comerciales.
- Disponibilidad y gestión de los recursos empresariales.
- Compatibilidad de la API.
- Definir los públicos objetivos o consumidores.
- Crear valor, canales y motivaciones para el uso de la API.
- Generar valor para el usuario final a través de la aplicación que utiliza la API.

2.8.3 CAPAS DE ARQUITECTURA API

A continuación, se detallará una serie de componentes que ayudan a la creación de una arquitectura API, según J (2020), quien manifiesta lo siguiente: Tabla 2.

Tabla 2**: Capas de arquitectura API.**

Capas	Definición
Gestión de la información	Funcionan como centros de acopio de los datos. Las API se debe entregarse con inventarios en los tiempos reales para los cuales se requieren las bases de que los datos sean constantes, precisas y confiables.
Aplicación/Microservicios	Albergan las aplicaciones que se van ejecutando en las organizaciones.
Integración	Ayuda en las integraciones de la API para los accesos síncronos a múltiples servicios en los usuarios y dispositivos.
Interacción	En esta posición es donde se da las acciones reales.

Fuente: J (2020)

2.9 MICROSERVICIOS

Los microservicios son un enfoque arquitectónico y organizativo para el desarrollo de software, donde el software está compuesto por pequeños servicios independientes que se comunican a través de API bien definidas. La arquitectura de

microservicios hace que las aplicaciones sean más fáciles de escalar y más rápidas de desarrollar. Esto acelera el tiempo de entrega de las nuevas capacidades (AWS, 2023). Los microservicios son tanto un estilo de arquitectura como un modo de programar software.

Con los microservicios, las aplicaciones se dividen en sus elementos más pequeños e independientes entre sí. Los microservicios son elementos independientes que funcionan en conjunto para llevar a cabo las mismas tareas. Es un elemento fundamental de la optimización del desarrollo de aplicaciones hacia un modelo nativo de la nube (Redhat, 2024).

2.10 DISEÑO DE ARQUITECTURA DE MICROSERVICIOS.

Una arquitectura de microservicios compone un conjunto de servicios con funcionalidades pequeñas y autónomas dentro de un contexto delimitado (DevOps, s.f.). Hay dos componentes que constituyen la arquitectura de microservicios: Administración e implementación y la Puerta de enlace de API.

Administración e implementación: este componente coloca los servicios en los nodos, localiza los errores, reequilibra los nodos y el servicio; además, tiene una tecnología estándar.

Puerta de enlace de API: este componente es la entrada de los clientes. Esta puerta reenvía la llamada a los servicios correspondientes en el backend. Los servicios utilizan protocolos de mensajería para un servicio web, como AMQP (Protocolo Avanzado de Colas de Mensajes). Además, la puerta de enlace de API realiza otras funciones, como la autenticación, la terminación SSL, el equilibrio de carga y los registros.

Para crear una arquitectura de microservicios, se realiza lo siguiente:

Análisis de dominios: Dentro de este análisis, hay tres pasos a seguir.

1. Análisis de dominios para modelar microservicios: Previamente a la codificación, se necesita una visión general del sistema que se va a construir. En el diseño basado en dominios, se inicia modelando el dominio empresarial y se crea un modelo de dominio. El modelo de dominio es un modelo abstracto del ámbito empresarial. Extrae y organiza el conocimiento del dominio y proporciona un lenguaje común para los desarrolladores (DevOps, 2023). Es decir, para emplear el análisis de dominio, se debe tener en claro las necesidades empresariales. Se puede realizar un diagrama determinando las funciones empresariales y sus conexiones; de esta manera, se ubica donde la aplicación se deberá integrar con sistemas externos.

2. Diseño basado en dominios táctico para diseñar microservicios: Definir con precisión los modelos de dominio. Los patrones tácticos se aplican dentro de un único contexto determinado (Migratel, s.f.). Es decir, en esta estrategia se trabaja con contextos empresariales delimitados, siguiendo los siguientes patrones tácticos: entidades, objetos de valor, agregados, servicios de aplicación y de dominio, y finalmente, eventos de dominio.

3. Identifique los límites de los microservicios: Se empieza con un contexto delimitado, que marca el límite del modelo de dominio. Se observan los agregados, que derivan de los requisitos empresariales, y además se deben tomar en cuenta los requisitos no funcionales.

Funcionamiento en producción: Dentro de este paso, se deben tener operaciones sólidas, como las siguientes.

CI/CD para arquitecturas de microservicios: El CI/CD son varios procesos, como la Integración continua, la Entrega Continua y la Implementación continua.

Como objetivos de un proceso de CI/CD tenemos:

- Cada equipo puede compilar e implementar los servicios que posee de forma independiente, sin afectar ni interrumpir a otros equipos.
- Previos a liberar en producción una nueva versión del servicio, se implementa en entornos de desarrollo para promover a un entorno de pruebas y control de calidad.

- Se aplican pruebas de calidad en cada fase, con pruebas unitarias, cobertura de código, de seguridad y rendimiento, según el ambiente que corresponda.
- Se puede priorizar una actualización a una nueva versión del mismo servicio, en paralelo con la versión anterior.
- Tiene fuertes directivas para control de acceso.
- Para las cargas de trabajo, se implementan imágenes de contenedor que son altamente confiables (Azure, s.f.).

Creación de una canalización de CI/CD para microservicios en Kubernetes: Los objetivos gestionados con CI/CD para microservicios alojados por Kubernetes son: el equipo de desarrollo puede compilar e implementar sus servicios de manera independiente. Los cambios de código a desplegar en los ambientes que pasan el proceso de integración continua se implementan automáticamente en un entorno similar al de producción. Se aplican pruebas de calidad en cada fase de la canalización (AKS, s.f.).

Supervisión de los microservicios que se ejecutan en Azure Kubernetes Service (AKS) Los procesos recomendados para la aplicación de microservicios son telemetría, supervisión del estado del clúster, métricas, registro, registro estructurado y seguimiento distribuido (Monitor, s.f.).

2.11 DISEÑO DE LA ARQUITECTURA EMPRESARIAL

Se trabaja con dos escenarios: racionalización y transformación digital. Se eligió diseñar una arquitectura empresarial utilizando TOGAF 9.2, basada en los estándares ArchiMate 3.1 para el modelado.

TOGAF 9.2

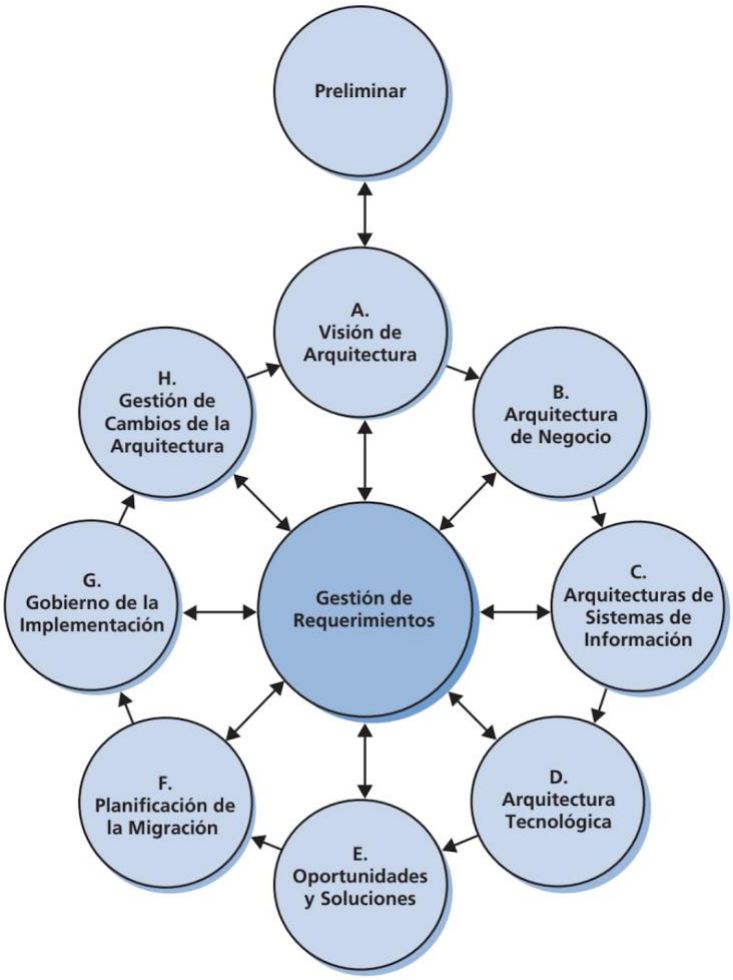
El marco TOGAF es un estándar de la industria para el desarrollo, implementación y administración de la arquitectura empresarial. También describe puntos de vista técnicos y modelos de referencia que ayudan a reducir gastos, mejorar el rendimiento de las inversiones en tecnología, normalizar procesos de negocio y

mejorar la comunicación; todo esto con el objetivo de mejorar la toma de decisiones empresariales. La figura 5 muestra el método de desarrollo de arquitectura (ADM).

ArchiMate 3.1

El estándar ArchiMate fue diseñado desde cero para que los arquitectos comuniquen su trabajo de forma coherente y eficaz. Esto se logra gracias a un conjunto sólido de conceptos y a la idea de puntos de vista arquitectónicos, tomados de la norma ISO/IEC 42010.

Figura 5: ADM El Ciclo del Método de Desarrollo de la Arquitectura.



Fuente: Josey, A. (2013).

Es decir, un punto de vista de arquitectura es un grupo de acuerdos que se pueden utilizar para producir una vista de arquitectura. Una vista podría ser un esquema, un catálogo, una matriz u otra forma de describir un subconjunto de su

arquitectura, con el fin de responder a una preocupación conocida de un conjunto identificado de partes interesadas.

El estándar ArchiMate es abierto y tiene una gran aceptación. Es un lenguaje gráfico que permite retratar y visualizar el modelo de arquitectura empresarial, soportando el ciclo completo del diseño de la arquitectura. La especificación ArchiMate 3.0.1 se basa en un lenguaje central dirigido a la descripción de negocios, sistemas de información, tecnologías de TI, arquitecturas de tecnología física y sus interrelaciones.

La especificación ArchiMate 3.0.1 también incluye elementos para modelar la estrategia empresarial y tecnológica, las motivaciones que impulsan el desarrollo de la arquitectura, así como planes para la implementación y migración de la arquitectura.

Racionalización

La inflexibilidad de la arquitectura de aplicaciones de la actual solución empresarial dificulta la adaptación a los cambios en las condiciones comerciales, debido a la obsolescencia de los componentes que conforman la aplicación.

Esto es, en parte, el resultado del crecimiento de las empresas que requieren estos servicios financieros, lo que ha llevado a un aumento de datos y a una superposición funcional. Esto se debe a la concentración de los requerimientos por los mismos canales de atención, así como a la integración de aplicaciones punto a punto utilizando una variedad de formatos y métodos de datos.

Transformación digital

Además de la necesidad del negocio, la institución financiera, como la mayoría de las empresas, está atravesando un cambio estructural para incorporar la transformación digital en sus procesos. Esto implica modernizar los equipos de desarrollo, incorporar nuevas metodologías y realizar otros cambios en consecuencia para apoyar a las empresas en su transformación. Se plantea una

nueva estrategia de integración digital con sus clientes, que emplea una combinación de Big Data y comunicación en línea mediante APIs.

La entidad financiera necesita acelerar las integraciones con las empresas, proveyendo herramientas que les permitan a estas autogestionarse e integrarse con sus propios sistemas ERP o similares.

Con este fin, para los productos financieros afiliados a las empresas, la entidad financiera tiene la intención de utilizar una comunicación directa entre sus sistemas internos, como el ERP, y los servicios financieros requeridos. En última instancia, esto puede resultar en servicios financieros en tiempo real, donde los clientes reciban respuestas directas sobre el estado de sus solicitudes y puedan tomar las acciones necesarias.

DESARROLLO DEL PROYECTO

Para el presente trabajo, tomamos como referencia la oferta actual de las instituciones financieras que trabajan en la arquitectura de aplicación web para el segmento de empresas. Estas instituciones cuentan con un sistema que realiza la gestión y apoya el negocio. Sin embargo, la inflexibilidad de la arquitectura monolítica de aplicaciones de la actual solución empresarial dificulta la adaptación a los cambios en las condiciones comerciales.

Para el desarrollo de este proyecto se plantean dos instancias. La primera se enfoca en el diseño de la arquitectura empresarial, con el fin de determinar una solución acorde a las necesidades del negocio de una entidad financiera en proceso de transformación digital en Ecuador. La segunda consiste en proponer un diseño de la solución graficado con el estándar C4.

3.1 MÉTODO DE DESARROLLO DE LA ARQUITECTURA (ADM) DE TOGAF

Para este trabajo se ejemplifican las siguientes fases, las cuales son propuestas por el Método de Desarrollo de la Arquitectura (ADM) de TOGAF. A continuación, se presenta la tabla 3: Método de Desarrollo de la Arquitectura (ADM) de TOGAF.

Tabla 3: **Método de Desarrollo de la Arquitectura (ADM) de TOGAF**

Fases	*Definición*
Fase A:	Visión de la Arquitectura
Fase B	Arquitectura de Negocio
Fase C:	Arquitectura de Sistemas de Información
Fase D	Arquitectura Tecnológica

Fuente: Propia

3.1.1 FASE A: VISIÓN DE LA ARQUITECTURA

PARTES INTERESADAS (STAKEHOLDER)

Se identifican las principales partes interesadas en el compromiso de la arquitectura y sus preocupaciones. El estándar ArchiMate las denomina impulsores internos. TOGAF define una matriz de mapa de partes interesadas para representar esto. La figura 6 muestra dichas partes interesadas, donde se identifican dos stakeholders: la Entidad Financiera y sus Clientes Empresariales (actuales y potenciales). Se destacan sus principales preocupaciones, siendo la más relevante satisfacer a sus clientes en el pago de nómina a través del sistema financiero.

Para la Entidad Financiera, los representantes del segmento de empresas desean incrementar la transaccionalidad de pagos de nómina, proporcionando un canal adicional de comunicación. Para lograrlo, es fundamental la afiliación de nuevas empresas.

Donde:

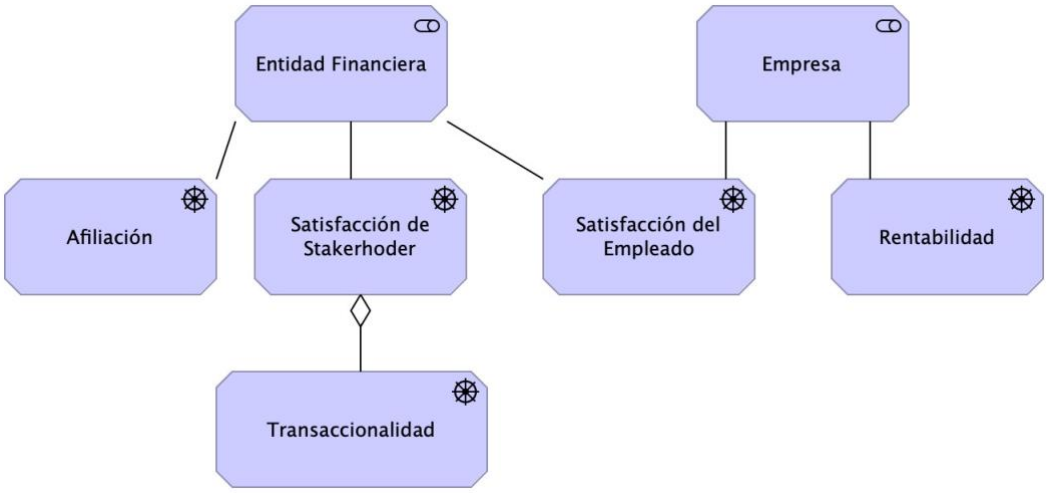
Stakeholder: Simboliza un individuo, equipo u organización que representa sus intereses en los resultados de la arquitectura.

- Entidad Financiera.
- Empresa.

Identificamos el Conductor que representa una condición interna o externa que justifica a la entidad financiera definir sus objetivos e implementar los cambios necesarios para alcanzarlos son:

- Afiliación de nuevas empresas,
- Satisfacción de Stakerhoder,
- Satisfacción del Empleado de la empresa afiliada,
- Rentabilidad del servicio y Transaccionalidad alta

Figura 6: Stakeholder o partes interesadas de este proyecto.



Fuente: Propia

OBJETIVOS COMERCIALES ASOCIADOS A LA RENTABILIDAD

La figura 7 muestra los objetivos comerciales asociados con la rentabilidad que las partes interesadas desean alcanzar. Este análisis, desde el punto de vista de la entidad financiera y basado en el sistema tradicional de gestión del pago de nómina, revela el objetivo comercial de aumentar constantemente las transacciones procesadas, generando mayor rentabilidad.

El aumento de rentabilidad en el pago de nómina está directamente relacionado con la afiliación de nuevas empresas, sin afectar la disponibilidad del servicio para las empresas ya afiliadas. Además, se identifica la necesidad de un proceso de afiliación ágil que permita transaccionar lo más pronto posible, mientras de forma paralela se realiza la afiliación a la plataforma tradicional, lo que permitirá acceder a más productos ofrecidos por la entidad financiera. Este proceso es transversal, promoviendo una mejora continua en todos los productos involucrados, con el objetivo de consolidar una plataforma única para las empresas.

Donde:

Conductor:

- El conductor es el aumento de la transaccionalidad.

Evaluación: Representa el resultado del análisis de una situación de la empresa respecto a algún factor determinante:

- Empresas que, al finalizar el proceso de afiliación, deciden optar por competidores con menor carga operativa.
- Empresas medianas o pequeñas en crecimiento con necesidades financieras moderadas o bajas.
- Procesos internos ineficientes debido al alto volumen de información.
- Proceso de integración estandarizado.
- Portal de autogestión para facilitar la integración.

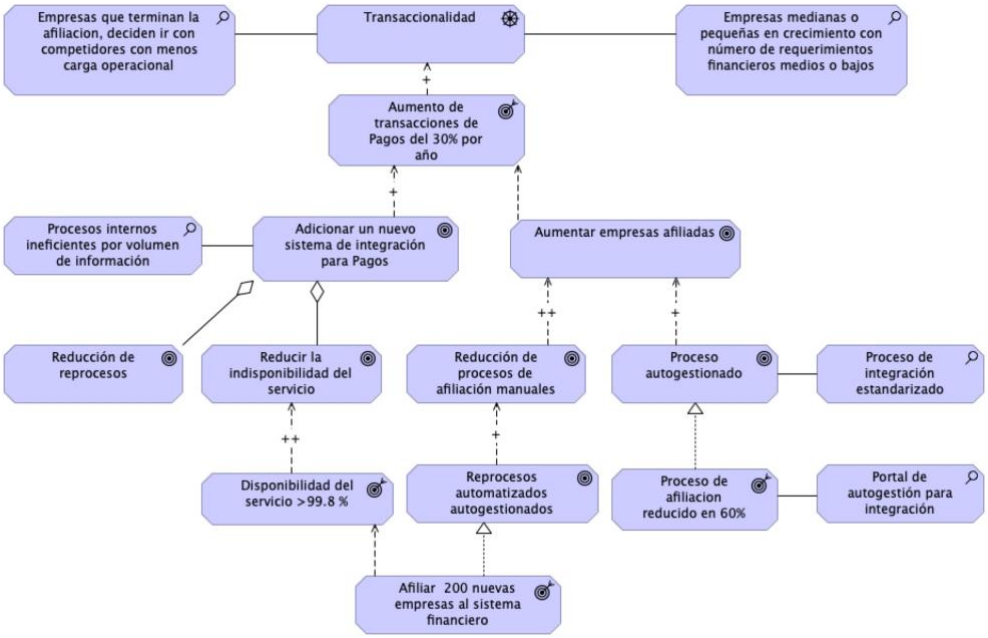
Resultado: Representa el efecto o consecuencia de una situación determinada:

- Aumento de transacciones de pagos anuales.
- Reducción en el proceso de afiliación.
- Alta disponibilidad del servicio.
- Afiliación de nuevas empresas al sistema financiero.

Metas: Representa una declaración de intención, dirección o estado final deseado a nivel organizacional:

- Implementar un nuevo sistema de integración para pagos.
- Incrementar el número de empresas afiliadas.
- Reducir los reprocesos.
- Disminuir la indisponibilidad del servicio.
- Automatizar el proceso de afiliación, reduciendo la necesidad de procedimientos manuales
- Promover la autogestión en los procesos.
- Automatizar y autogestionar los reprocesos.

Figura 7: Objetivos comerciales asociados con la rentabilidad de la entidad financiera.



Fuente: Propia

PRINCIPIOS DE ARQUITECTURA.

El diagrama de Principios permite organizar los principios más relevantes para el diseño del pago de nómina en línea. Estos principios pueden influirse entre sí, de forma positiva o negativa, y permiten organizar las relaciones entre los principios y sus metas.

En la figura 8 se muestran los principios, sus dependencias y los objetivos asociados. Se identifican dos metas: incrementar las empresas afiliadas e incluir un nuevo sistema de integración ágil entre la empresa y la entidad financiera para el servicio de pago de nómina.

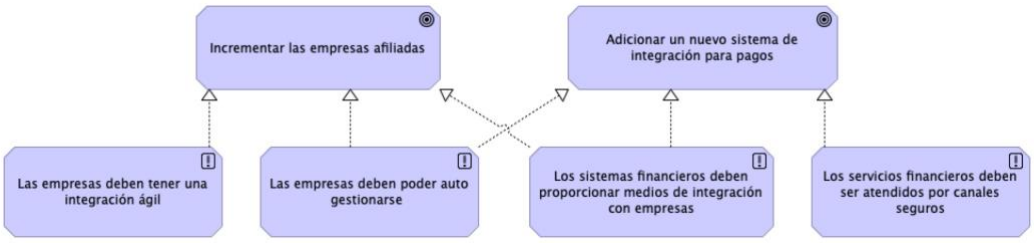
Metas:

- Incrementar las empresas afiliadas.
- Adicionar un nuevo sistema de integración para pagos.

Los principios identificados están estrechamente relacionados con los objetivos y definen las propiedades generales que se aplican al sistema en el contexto actual:

- Las empresas deben tener una integración ágil.
- Las empresas deben poder autogestionarse.
- Los sistemas financieros deben proporcionar medios de integración con las empresas.
- Los servicios financieros deben ser atendidos a través de canales seguros.

Figura 8: Principios de arquitectura.



Fuente: Propia

REFINAMIENTO OBJETIVOS ESTRATÉGICOS.

Los objetivos y principios son la base para definir los requisitos concretos. El refinamiento de objetivos permite modelar metas generales y transformarlas en objetivos más específicos o concretos. Las metas, respaldadas por sus principios, son necesarias para alcanzar el objetivo principal: incrementar las empresas afiliadas al servicio de pago de nómina.

La figura 9, Refinamiento de Objetivos – Estrategia de Afiliación, muestra el refinamiento del objetivo estratégico, junto con los requisitos para la afiliación de nuevos clientes en el servicio de pago de nómina. Se describe cómo las metas se relacionan con sus principios y cómo estos contribuyen e impulsan el logro del objetivo estratégico.

Donde:

Metas:

- Incrementar las empresas afiliadas.

- Reducir los procesos de afiliación manuales.
- Promover un proceso autogestionado.
- Disminuir los reprocesos.

Principios:

- Las empresas deben tener una integración ágil.

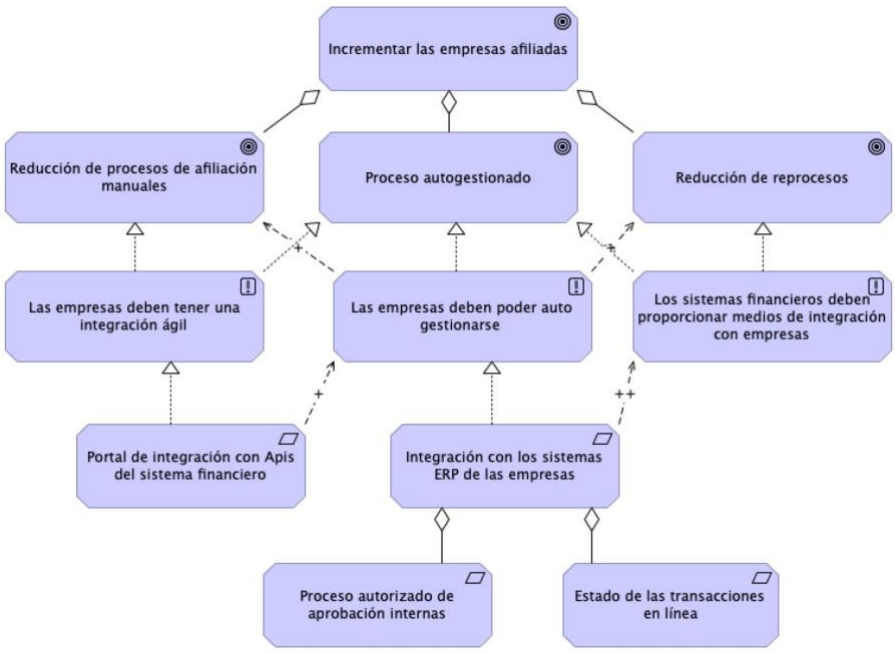
- Las empresas deben poder autogestionarse.
- Los sistemas financieros deben proporcionar medios de integración con las empresas.

Requisitos:

Se identifican los requisitos, los cuales representan la declaración de una necesidad que define una propiedad aplicable al sistema:

- Portal de integración con APIs del sistema financiero.
- Integración con los sistemas ERP de las empresas.
- Proceso autorizado de aprobación interna.
- Estado de las transacciones en línea.

Figura 9: Refinamiento, Objetivo Estrategia Afiliación



Fuente: Propia

ESTRATEGIA A LARGO PLAZO

Visión a largo plazo además de la necesidad de afiliación a corto plazo. A largo Plazo se ha definido una estrategia de intimidad digital del cliente, que combina ERP o sistemas internos con el sistema financiero, además de una plataforma con una arquitectura orientada a la nube, con interconexiones a plataformas especializadas multicore, centralizando los productos y servicios financieros, con afiliación ágil, auto gestionado que le permita administrar los productos que desea incorporar a sus empresas.

Se tiene la intención de utilizar datos más detallados de los clientes, para mejorar la interacción, la satisfacción del cliente y personalizar en sus propios sistemas sus necesidades con el sistema financiero. Las empresas se integrarán con un sistema estándar para la integración con el sistema financiero a través de Apis, que pueden llegar a ser monetizados.

La estrategia permite modelar una visión general de los cursos de acción elegidos o considerados por el sistema financiero, las capacidades y los recursos que los apoyan, los resultados previstos y cómo estos contribuyen a los objetivos y los impulsores de la organización.

La figura 10 muestra la estrategia a largo plazo del objetivo estratégico, para la afiliación de nuevos clientes para el pago de nómina y los deseables que en futuras entregas de acuerdo con la planificación transversal de los actores del segmento empresas lo prioricen considerando los objetivos con entidad financiera que tenga en cada ciclo. Donde:

Stakeholder

- Entidad Financiera.

Conductor

- Transaccionalidad

Evaluación

- Empresas que terminan la afiliación, deciden ir a competidores con menos carga operacional y
- Procesos internos ineficientes por volumen de información.

Meta

- Reducción de procesos de afiliación manuales,
- Incrementar las empresas afiliadas,
- Bancarizar nuevas empresas medianas y pequeñas,
- Proceso auto gestionado y
- Aumentar clientes con productos financieros

Curso de Acción: Es plan para distribuir capacidades y recursos de la entidad financiera para lograr el objetivo.

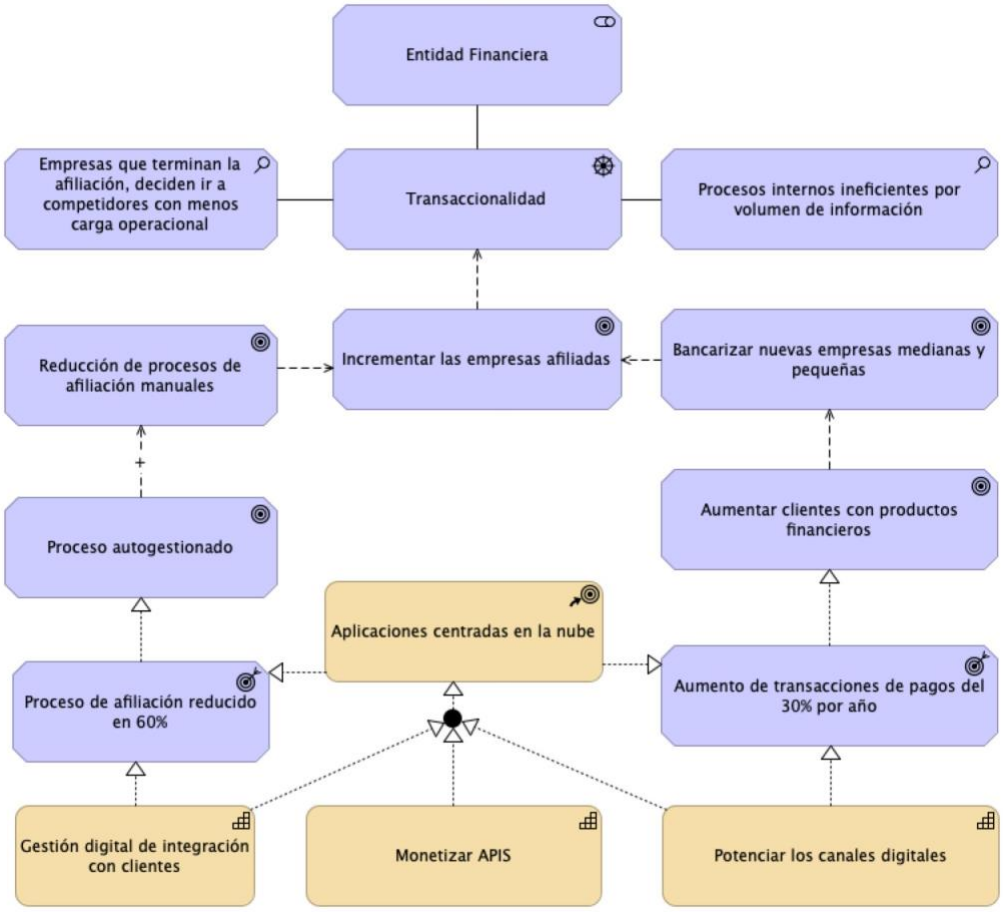
- Aplicaciones centradas en la nube

Capacidad: Las capacidades identificadas que representa una habilidad o capacidad del sistema.

- Gestión digital de integración con clientes,
- Monetizar APIS y
- Potenciar los canales digitales

En última instancia, esta nueva estrategia, por supuesto, también debería contribuir a los principales impulsores de la organización, como se describe en parte en la figura 6 stakeholder. Las relaciones entre la estrategia, las capacidades, el resultado previsto y los impulsores de las partes interesadas se muestran en la figura 10 (estrategia largo plazo). La estrategia de aumentar la transaccionalidad requiere que la entidad financiera desarrolle una serie de nuevas capacidades y recursos, incluida la gestión digital de nuevos clientes y potenciar los canales digitales.

Figura 10: Estrategia a largo plazo



Fuente: Propia

CONCEPTO DE SOLUCIÓN

La figura Concepto de Solución destaca los aspectos más importantes de la arquitectura objetivo, mostrando los requisitos y resultados derivados tanto de la afiliación de nuevas empresas como del uso de tecnologías orientadas a la nube, para potenciar los canales digitales y bancarizar nuevos clientes.

- La integración con los sistemas ERP o sistemas propios de las empresas, a través de un API Manager, les permitirá autogestionarse en sus desarrollos de integración y pruebas.
- La integración con APIs reducirá los procesos manuales en herramientas externas, lo que generará un proceso de mantenimiento de los sistemas y de la operación más eficiente.

- La automatización de la generación de los mensajes de comunicación entre las empresas y la entidad financiera reducirá los costos de retrocesos y la gestión de incidentes.
- La disponibilidad del servicio en línea para autogestionarse en cualquier momento mejorará la experiencia del cliente.
- La automatización del back-office integrado reemplazará las aplicaciones de back-office separadas para las diferentes líneas de negocio, lo que permitirá reducir y eliminar estas dependencias, mejorando la calidad de los datos manejados por los sistemas directamente en las empresas y sus sistemas internos.
- Bancarizar empresas permitirá a estas y a sus clientes acceder a productos financieros que potenciarán su crecimiento.

La figura 11 indica conceptualmente la solución para cumplir objetivo estratégico para la afiliación de nuevos clientes para el pago de nómina. Donde:

Resultado

- Aumento de transacciones de pagos por año.
- Afiliar nuevas empresas al sistema financiero.

Requisito

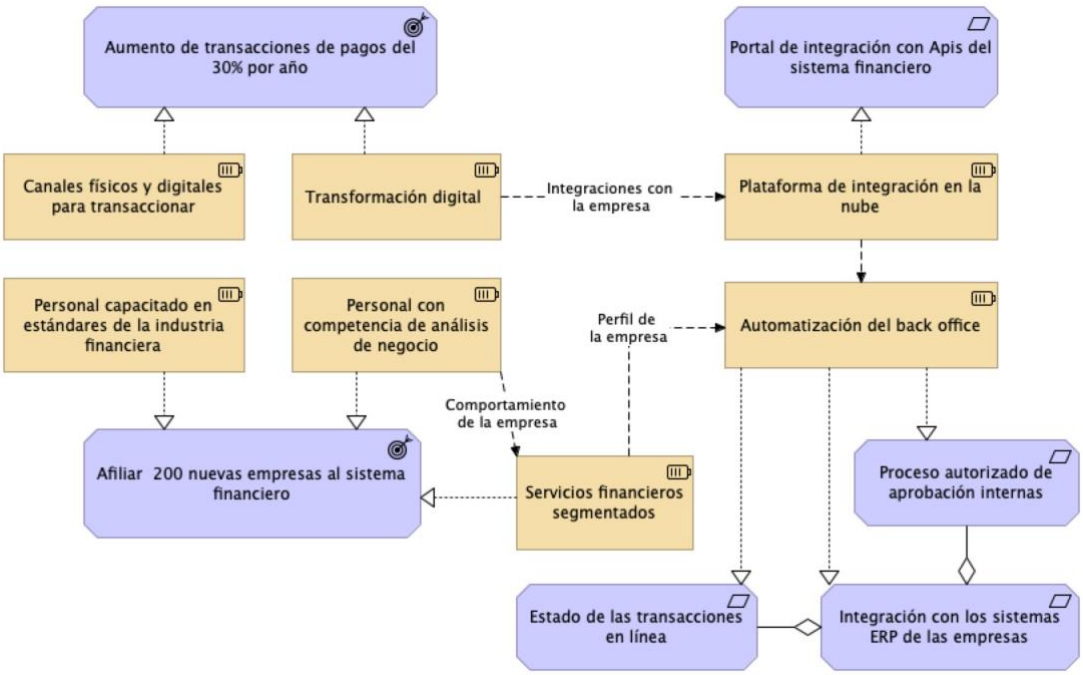
- Portal de integración con Apis del sistema financiero.
- Proceso autorizado de aprobación internas.
- Estado de las transacciones en línea.
- Integración con los sistemas ERP de las empresas.

Recurso representa un activo que pertenece o está controlado por la entidad financiera.

- Canales físicos y digitales para transaccionar.
- Transformación digital.
- Plataforma de integración en la nube.

- Personal capacitado en estándares de la industria financiera.
- Personal con competencia de análisis de negocio.
- Automatización del back office.
- Servicios financieros segmentados.

Figura 11: Concepto de la solución**.**



Fuente: Propia

3.1.2 FASE B: ARQUITECTURA EMPRESARIAL

En la Fase B (Arquitectura Empresarial) del TOGAF ADM, el lenguaje ArchiMate permite expresar y relacionar la estructura organizativa, los productos, los servicios, las funciones, los procesos y la información de la entidad financiera del segmento empresas. La arquitectura empresarial proporciona el contexto necesario para las arquitecturas de datos, aplicaciones y tecnología.

ESTRUCTURA DE ORGANIZACIÓN

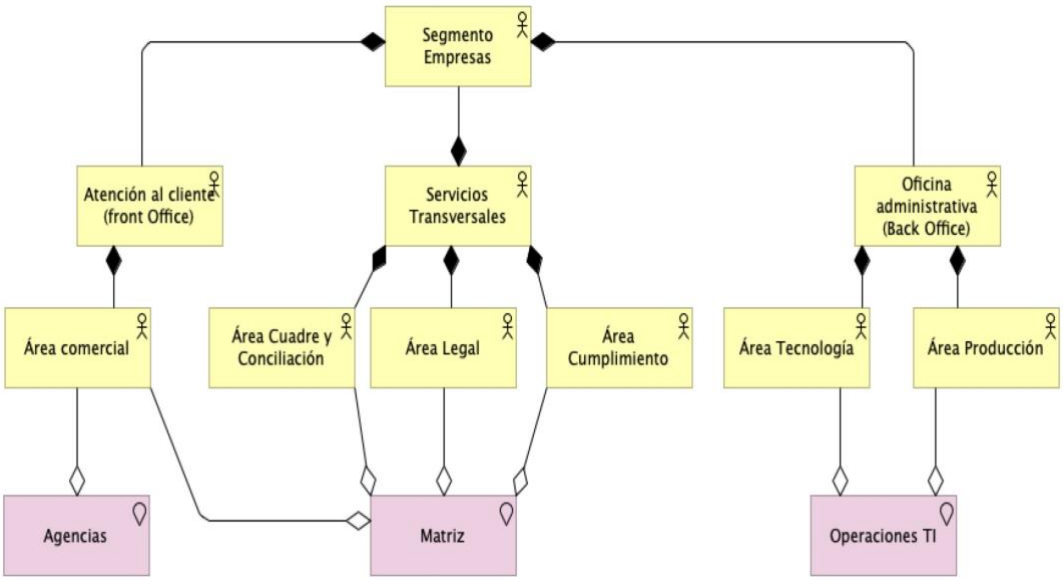
Se centra en la organización de una empresa, comercio, red de empresas o entidades similares. Los modelos pueden presentarse en forma de diagramas de bloques anidados o, de manera más tradicional, como organigramas. Esta

representación es útil para identificar responsabilidades y competencias dentro de una institución. En TOGAF, este modelo se conoce como el diagrama de descomposición de la organización.

La figura 12, "Estructura de la organización", muestra la organización de cualquier entidad financiera como referencia para el presente trabajo.

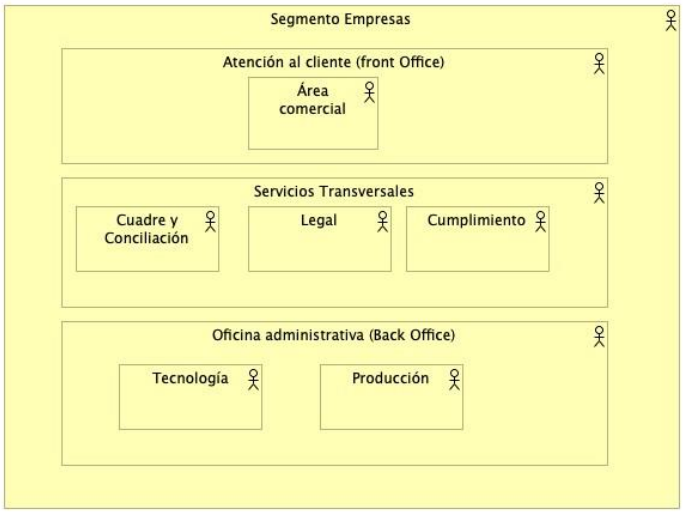
A menudo, la estructura de la organización se representa en forma de árbol. La figura 13, "Descomposición de la organización", ilustra la descomposición de la organización financiera. Esta vista refleja la estructura organizativa de alto nivel de la entidad financiera del segmento empresas, incluyendo sus principales ubicaciones y departamentos.

Figura 12: Estructura de la organización.



Fuente: Propia

Figura 13: Descomposición de la organización



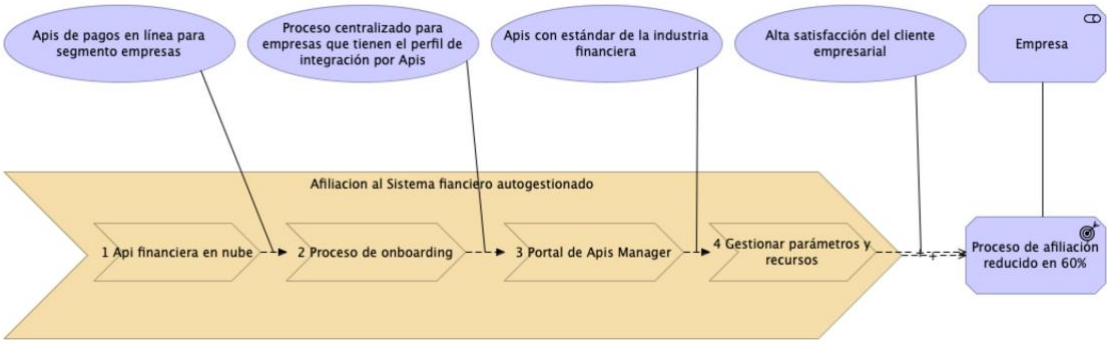
Fuente: Propia

FLUJO DE VALOR

Para expresar, a nivel estratégico, cómo una entidad financiera crea valor para sus clientes del segmento empresas, se utilizan flujos de valor. El principal flujo de valor de alto nivel está relacionado con la afiliación a productos financieros, como el servicio de pago de nómina.

Este flujo muestra las etapas de producción de valor, la contribución de cada etapa y el resultado final para el cliente. La figura 14, titulada "Flujo de valor", ilustra el proceso mediante el cual la entidad financiera genera valor para sus clientes a través de la integración al producto financiero de pago de nómina.

Figura 14: Flujo de valor.



Fuente: Propia

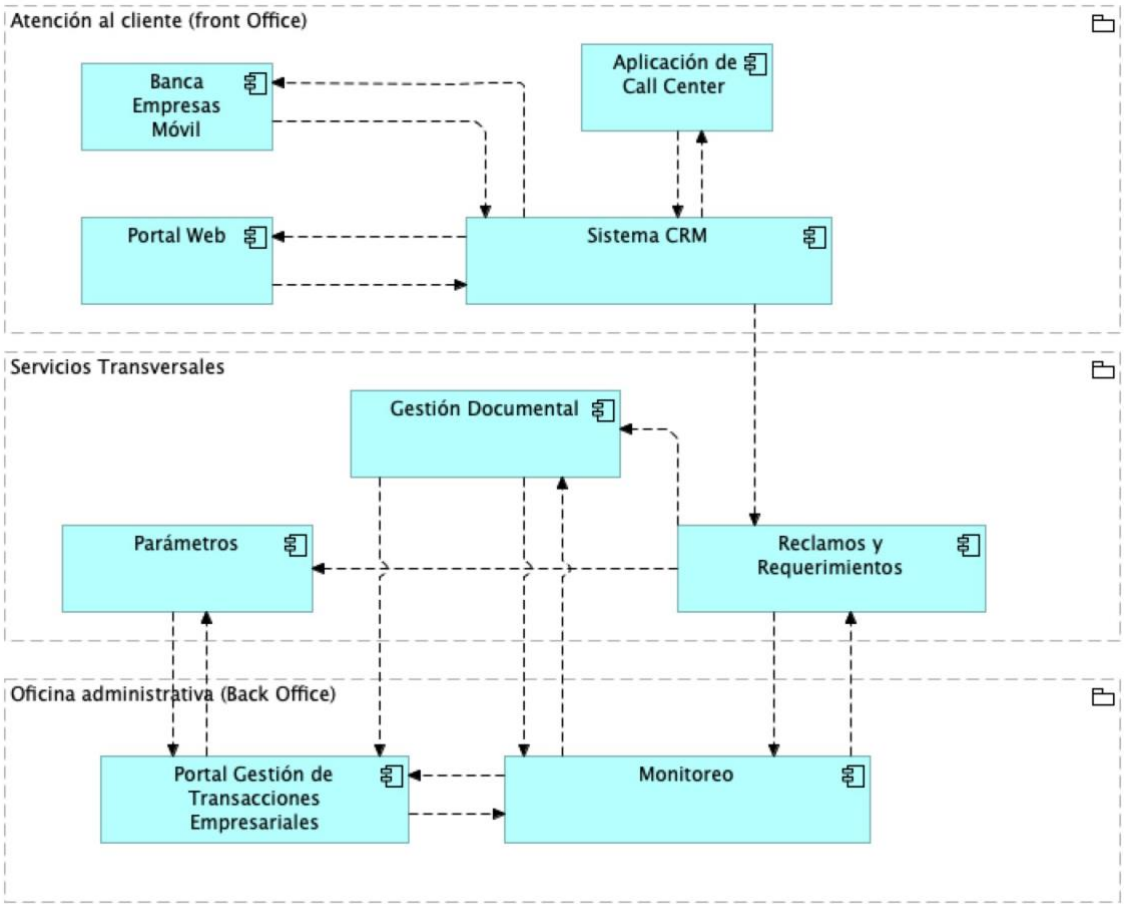
3.1.3 FASE C: ARQUITECTURAS DE SISTEMAS DE INFORMACIÓN (APLICACIÓN)

Con la necesidad de afiliar nuevas empresas, se abre un nuevo canal de integración en línea. Para ello, se utiliza el API Manager, que permite una autogestión para la interacción técnica con los servicios del sistema financiero, expuestos específicamente para este fin comercial: el pago de nómina.

Sin embargo, debido al enfoque de la administración en optimizar el onboarding, al mismo tiempo que mejora continuamente el rendimiento diario de cada división, la afiliación de la empresa no requiere realizar la integración al sistema actual, con el que las empresas existentes cuentan para su gestión diaria.

Por lo tanto, es importante implementar una solución para las empresas nuevas que están en crecimiento, con opciones acordes a su necesidad de transaccionalidad. Partimos de la arquitectura actual como línea base, que se describe en la figura 15 "Cooperación de aplicaciones (Línea base)". En el diagrama, se muestra la integración con los componentes de la entidad financiera para la integración al producto financiero de pago de nómina.

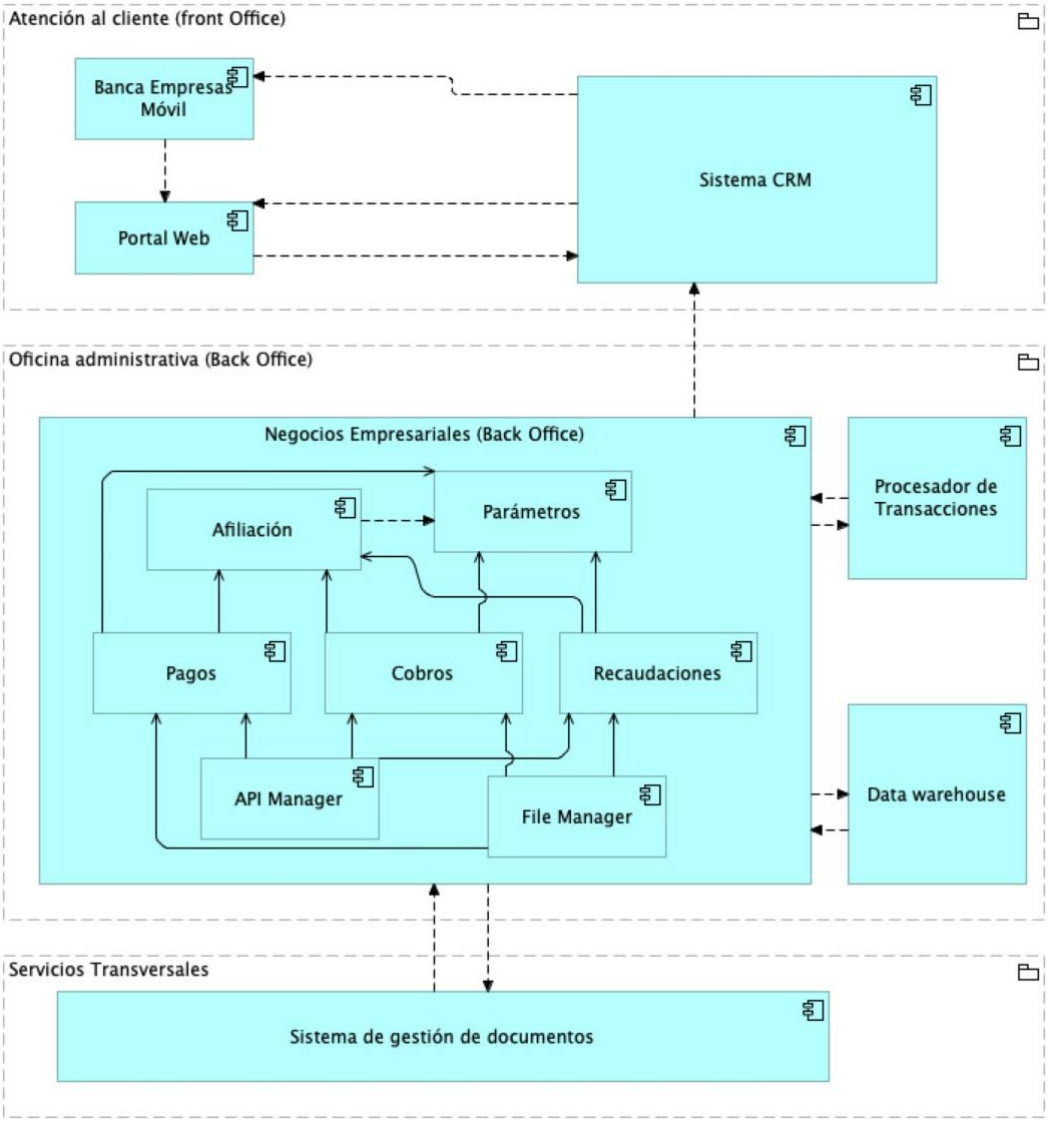
Figura 15: Cooperación de aplicaciones (Línea base).



Fuente: Propia

La arquitectura objetivo para el negocio se describe en la figura 16, "Cooperación de aplicaciones (Objetivo)". El diagrama expone la integración con los componentes de la entidad financiera, para incluir el producto financiero de pago de nómina, incorporándose a la arquitectura actual con todas sus fortalezas y procesos de mejora con los que cuenta la entidad financiera.

Figura 16: Cooperación de aplicaciones (Objetivo).



Fuente: Propia

COMPORTAMIENTO DE LA APLICACIÓN

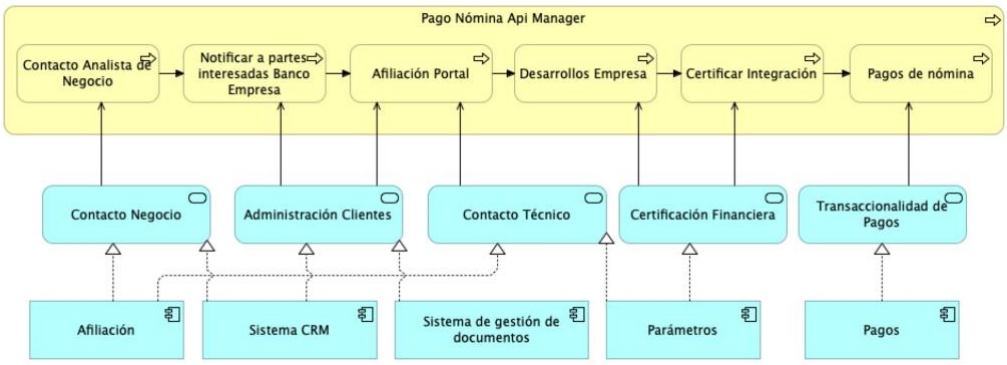
El comportamiento de la solución del pago de nómina con Api Manager, en el contexto de la afiliación de empresas nuevas, por un lado, y de los procesos de integración de nuevas opciones de negocio, por el otro.

Las nuevas empresas se afilian mediante un proceso de onboarding optimizado, en el que se autogestionan los métodos de integración para utilizar las APIs expuestas por el sistema financiero.

Las empresas que deseen afiliarse al producto de pago de nómina a través de APIs utilizarán el portal de API Manager para obtener la documentación necesaria, así como los mecanismos de seguridad requeridos para la integración directa con el

sistema financiero en sus diferentes etapas: desarrollo, calidad y producción. La figura 17, "Uso de la Aplicación API Manager Pago Nómina", muestra el flujo de uso de la aplicación con API Manager para el servicio financiero de pago de nómina, integrándose a la arquitectura actual con todas sus fortalezas y procesos de mejora que la entidad financiera posee.

Figura 17: Uso de la Aplicación Api Mánager Pago Nómina.

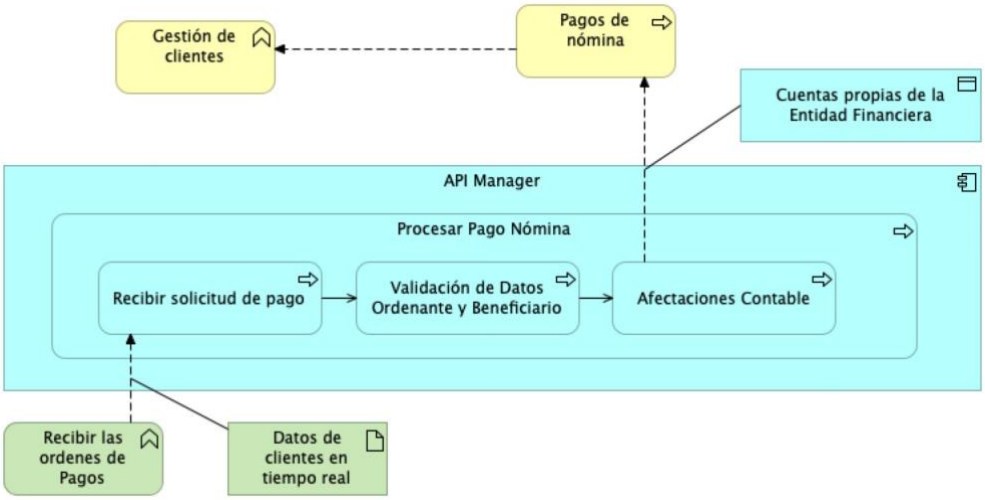


Fuente: Propia

El comportamiento del pago de nómina con un proceso en línea, integrado directamente a la empresa, generará nuevas oportunidades de negocio que permitirán a la empresa realizar pagos a través de sus sistemas, como el ERP, reduciendo procesos operativos en un entorno seguro. Esto contribuirá a que las empresas cumplan con sus trabajadores, y ofrecerá a la entidad financiera la oportunidad de bancarizar a nuevos clientes del segmento de personas, quienes tendrán la ventaja de unirse al sistema financiero y recibir sus salarios directamente en sus cuentas.

En la figura 18 expone el flujo de uso de la aplicación con API Manager para el servicio financiero de pago de nómina, con un proceso mejorado, objetivo de la institución financiera.

Figura 18: Uso de la Aplicación Api Mánager Pago Nómina Objetivo.



Fuente: Propia

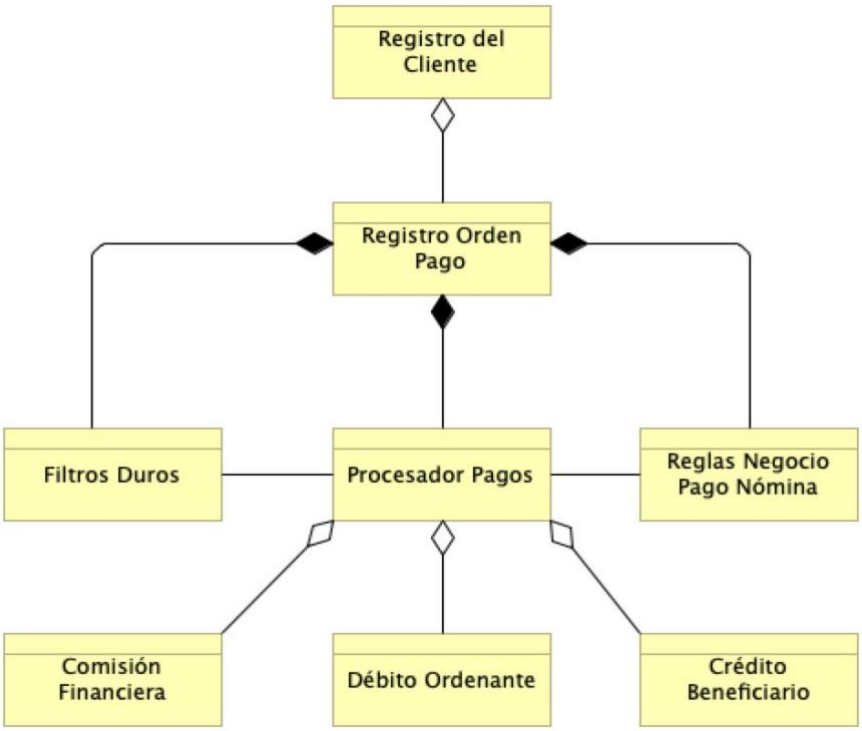
ARQUITECTURAS DE SISTEMAS DE INFORMACIÓN (DATOS)

La arquitectura de datos describe las principales relaciones entre los objetos de negocio conceptuales y los objetos de datos lógicos. Uno de los puntos de vista de datos definidos por el estándar TOGAF es el diagrama de datos lógicos.

Partiendo de la premisa de que una empresa ya es un cliente registrado en la entidad financiera dentro del segmento de empresas, se requiere que esta se afilie al producto de pago de nómina. Al tratarse de un servicio de pago de nómina, los empleados de dicha empresa deben registrar una cuenta en la entidad financiera, ya que el pago se realiza entre una cuenta de ordenante y una cuenta beneficiaria, ambas dentro de la misma entidad financiera.

Para otros productos o servicios financieros que la empresa pueda requerir, esta arquitectura propuesta permite integrar los procesos bajo las condiciones específicas del negocio financiero. Estos incluyen transacciones entre bancos (como transferencias interbancarias, con la intervención del Sistema Nacional de Pagos del Banco Central de Ecuador) y transacciones internacionales a través del sistema de transferencia SWIFT. Sin embargo, las integraciones adicionales al componente Procesador de Transacciones, que están fuera del alcance de este trabajo, no se detallan en esta descripción.

Figura 19: Estructura de información muestra los principales objetos comerciales.



Fuente: Propia

3.1.4 FASE D: ARQUITECTURA TECNOLÓGICA

El resultado de la fase será obtener los siguientes puntos:

- Arquitectura tecnológica: Vista de infraestructura.
- Comunicación de servicios: Recibir órdenes de pago.

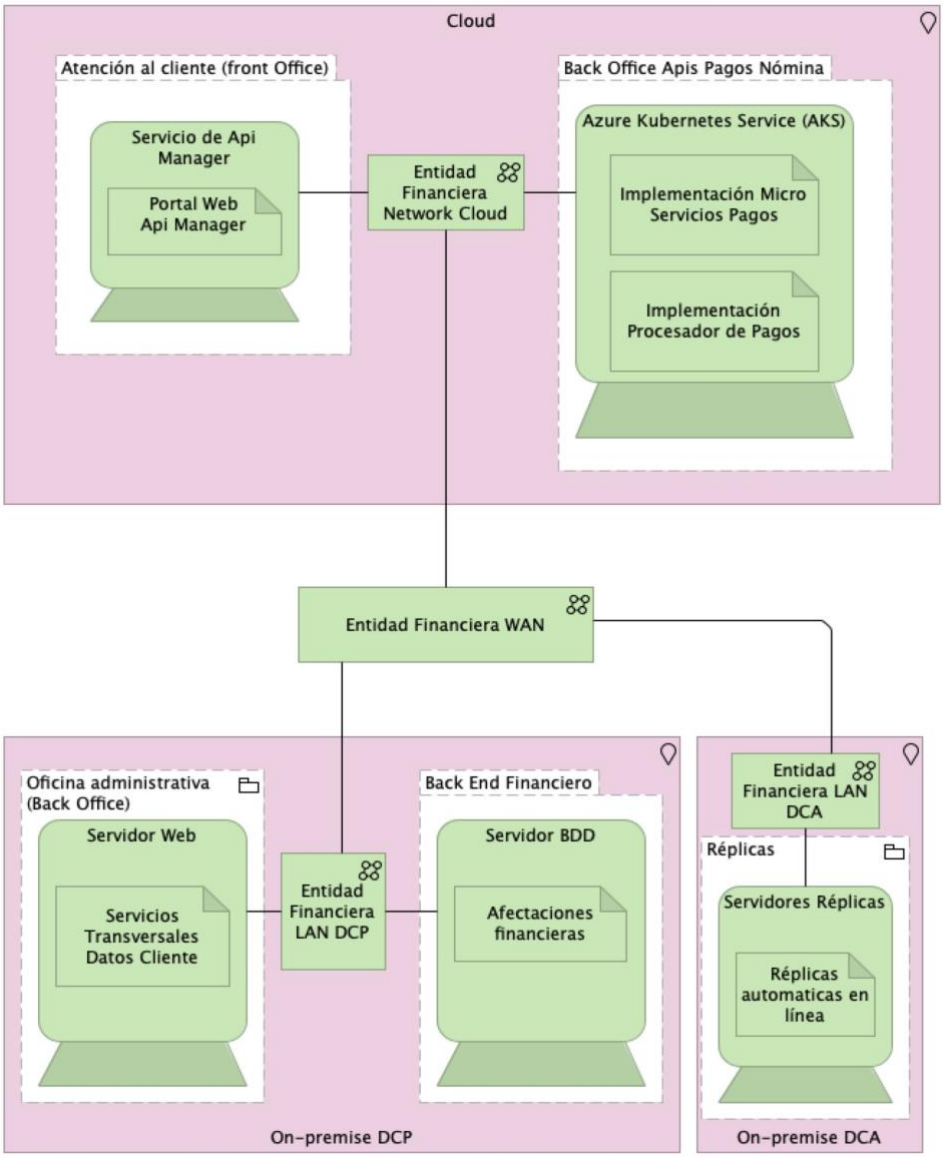
En cuanto al punto de vista de la infraestructura, se engloba tanto los elementos de infraestructura de software (sistemas operativos, bases de datos y middleware) como de hardware (dispositivos físicos), que son los que soportan la capa de aplicación. TOGAF representa este punto de vista mediante el diagrama de entornos y ubicaciones.

La entidad financiera cuenta con su propia infraestructura on-premise, que incluye servidores de aplicaciones legadas, bases de datos, middleware y el core bancario, con sus respectivas contingencias distribuidas en diferentes regiones geográficas.

Adicionalmente, dispone de infraestructura en la nube de Google y Microsoft, con diversos servicios que se ajustan según las necesidades del negocio y la capacidad técnica requerida.

La Figura 20 muestra la infraestructura (objetivo) de la entidad financiera que se toma como referencia.

Figura 20: Infraestructura (objetivo).



Fuente: Propia

INFRAESTRUCTURA

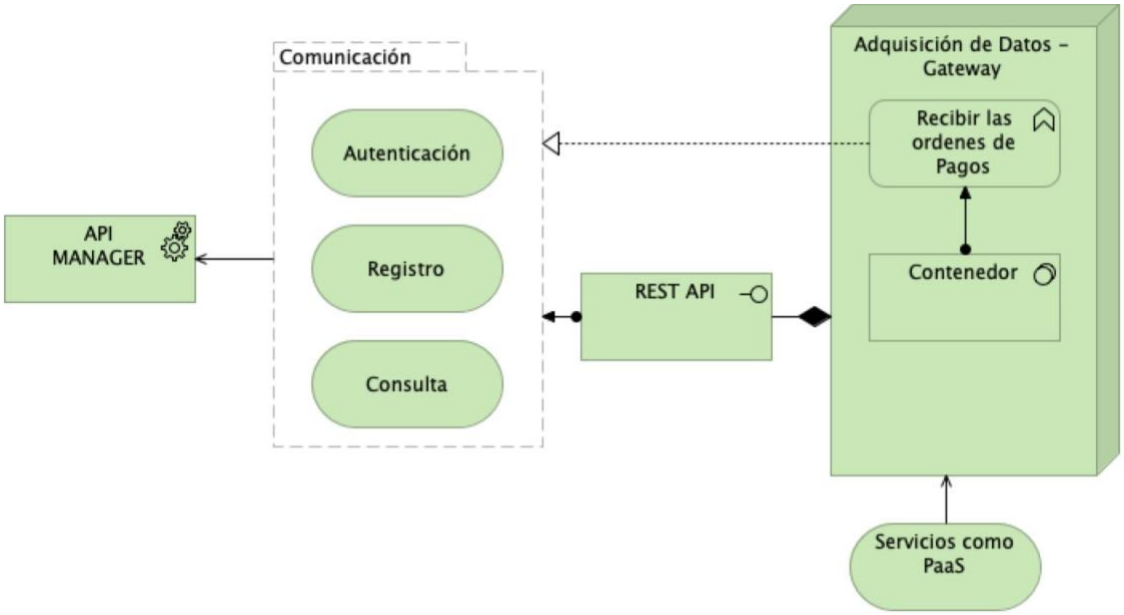
La implementación de la adquisición de datos para el pago de nómina con API Manager se basa en una arquitectura de microservicios. La empresa se integrará

con las APIs expuestas y documentadas en el portal de APIs de la entidad financiera. Las órdenes de pago de nómina se registran a través de una API REST.

También se utilizan servicios en la API para procesar las órdenes de los datos adquiridos. Para cada empresa afiliada, se ejecutará una instancia de la funcionalidad de adquisición de datos en un contenedor. La puerta de enlace en sí está respaldada por una plataforma como servicio (PaaS), que proporciona servicios para la implementación, la integración, la gestión del ciclo de vida del servicio, la seguridad, el equilibrio de carga, el almacenamiento y la virtualización.

La figura 21, "Comunicación Servicios Recibir Orden Pago Nómina", muestra la comunicación del servicio de pago de nómina en línea.

Figura 21: Comunicación Servicios Recibir Orden Pago Nómina.



Fuente: Propia

3.1.5 FASES E & F: OPORTUNIDADES, SOLUCIONES Y PLANIFICACIÓN DE LA MIGRACIÓN.

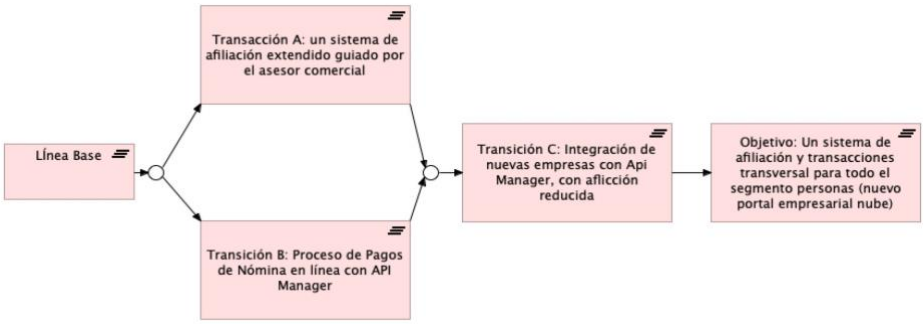
Para las fases E y F, el estándar TOGAF 9.2 incluye la arquitectura de transición, como una posible fase intermedia entre la arquitectura de línea de base y la arquitectura de destino, se muestran utilizando el punto de vista Migración. La migración involucra modelos y conceptos para definir la transición de una

arquitectura existente a una arquitectura deseada minimizando los riegos y mantener la continuidad de negocio en caso de sistemas o aplicaciones existentes o que tengan dependencias externas.

La criticidad de disponer de una plataforma intuitiva, auto gestionable e integral con alta disponibilidad requiere de una estrategia de migración, que permita a las empresas afiliadas continuar sus operaciones financieras y a las empresas nuevas afiliarse en un proceso optimizado y con tecnologías que lo respalden.

Por lo tanto, una arquitectura de transición reemplaza la afiliación dentro del portal empresarial existente con todos sus procesos y dependencias de infraestructura on-premises, por la integración con el Api Manager con una arquitectura orientada a microservicios en la nube, que después permita una integración con una nueva plataforma empresarial central orientada a la nube, que cumpla con los requerimientos de todo el segmento empresarial, Pymes y Corporativo.

Figura 22: Vista de migración.



Fuente: Propia

RESULTADOS Y DISCUSIÓN

Los diagramas generados para la arquitectura transitoria y la arquitectura objetivo se desarrollaron durante las fases B y E, respectivamente, del ciclo TOGAF. Estos diagramas permitieron identificar los requisitos de las partes interesadas, así como las limitaciones técnicas y de negocio que enfrenta la entidad financiera al tratarse de un servicio altamente transaccional.

Se identificó la necesidad de una arquitectura de transición que responda al incremento de la transaccionalidad, optimizando el proceso de afiliación e incorporando un modelo de negocio B2B para alcanzar los objetivos de las partes interesadas. Dentro del alcance de este trabajo, se propone el diseño de un sistema basado en APIs para el pago de nómina, enfocado en la afiliación de nuevas empresas.

Se ha decidido diseñar una arquitectura de APIs REST, aplicando el estándar BIAN. Para el modelado de la solución, se emplea el estándar C4, el cual proporciona diagramas jerárquicos de arquitectura de software que permiten abordar los objetivos planteados en este trabajo. El modelado incluirá los diagramas de Contexto, Contenedores y Componentes.

4.1 IMPLEMENTACIÓN BIAN

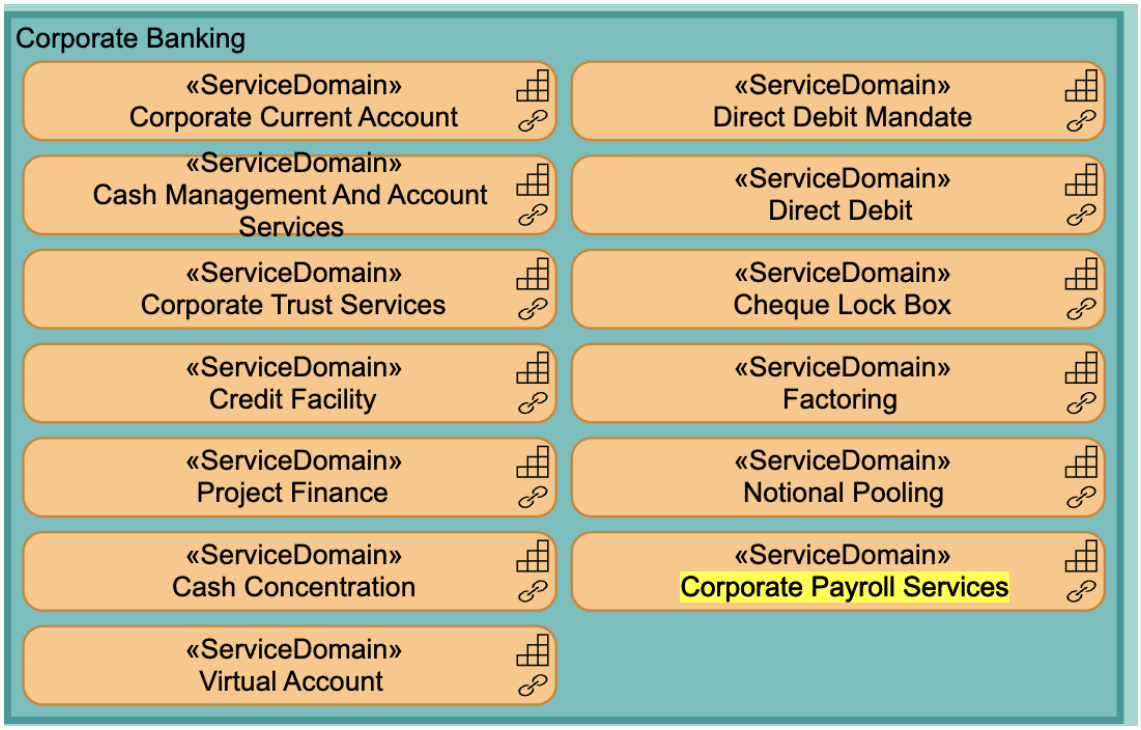
Se utiliza BIAN v12.0.0 para determinar una estructura de datos que facilite el intercambio de información a través de una API genérica. Esta API permite a las empresas afiliadas integrarse al servicio financiero para el pago de nómina, creando los mecanismos necesarios para la evolución del producto de la institución financiera hacia el mundo digital, con un nuevo modelo de intercambios y colaboración financiera.

El servicio de “Pago Corporativo de Nómina” forma parte del área de negocio "Operaciones y Ejecución" dentro del dominio empresarial de "Banca Comercial". Esto se ilustra en la figura 23 BIAN Service Domain Landscape versión 12.0 y se resume en la tabla 4. BIAN Service Landscape V12.0.

Figura 23: BIAN Service Domain Landscape version 12.0.

Fuente: BIAN(2025)

Figura 24: BIAN Service Landscape V12.0 Value Chain View



Fuente: BIAN(2025)

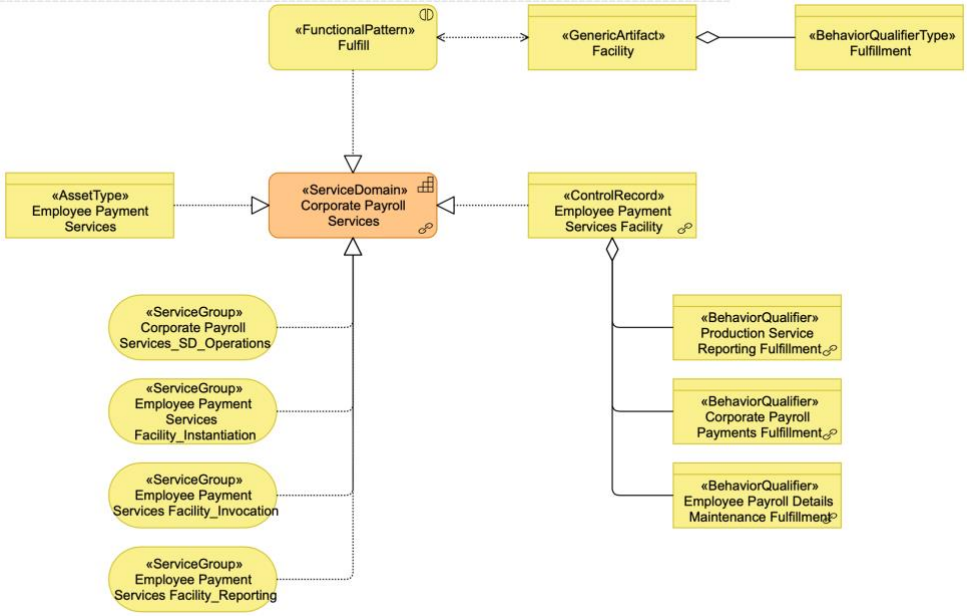
Tabla 4: **BIAN Service Landscape V12.0.**

Área De Negocio:	Operaciones y Ejecución
Dominio De Negocio:	Banca Comercial
Dominio de Servicio:	Servicios De Nómina Corporativa

Fuente: Propia

El estándar BIAN define el dominio de "Servicios de Nómina Corporativa" como aquel encargado del procesamiento de pagos de nómina para los empleados de clientes empresariales o corporativos. Este dominio gestiona el procesamiento de los pagos salariales netos de los empleados, con los detalles proporcionados por la empresa. La entidad financiera coordina los pagos hacia cualquier cuenta de empleado mantenida en la misma entidad y, en caso necesario, realiza pagos a cuentas en otras entidades financieras.

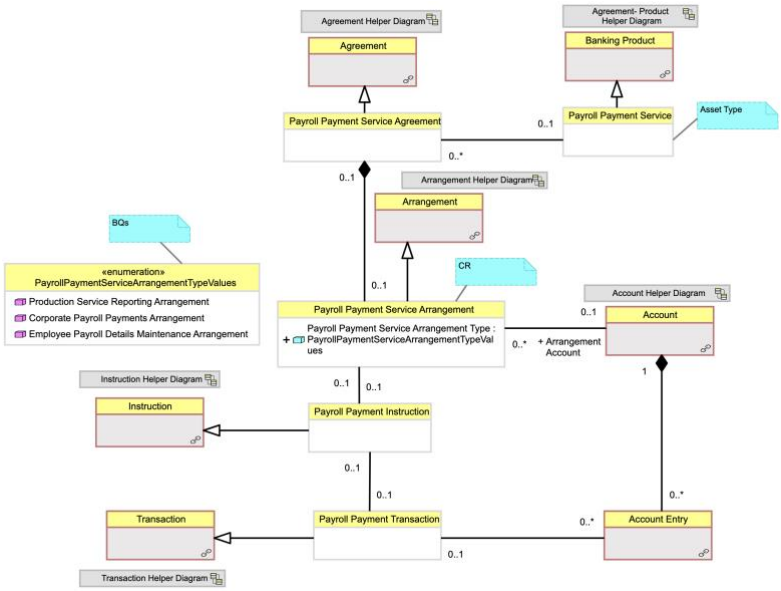
Figura 25: Descripción General del Servicio de Dominio de Nómina Corporativa



Fuente: BIAN(2025)

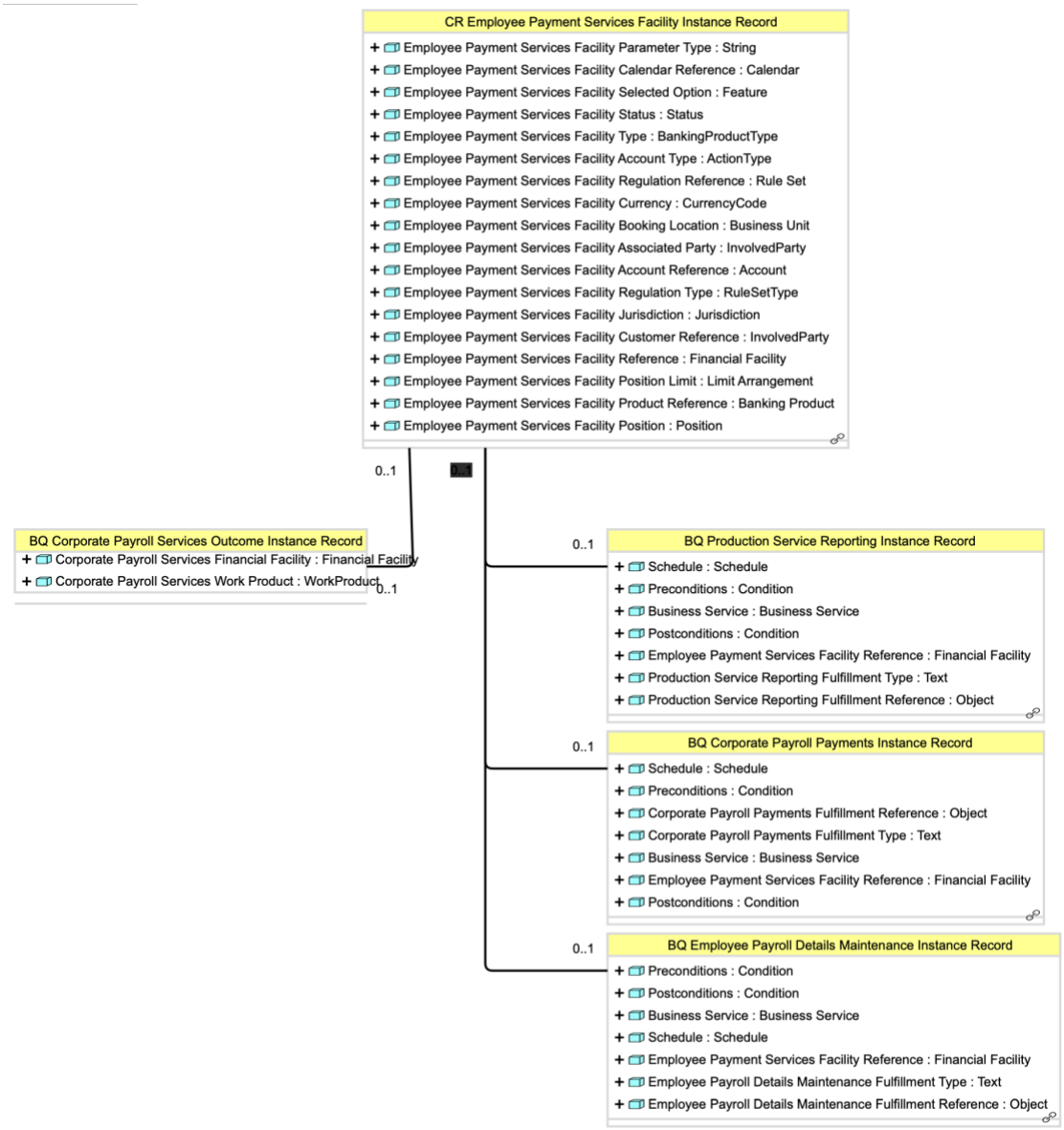
La definición de la estructura de datos y su nomenclatura se realiza a través de BIAN Semantic APIs. BIAN dispone de un portal de APIs donde se encuentra la documentación correspondiente, con especificaciones en OpenAPI, disponibles en la herramienta SwaggerHub. Estas especificaciones están definidas para cada dominio de servicio, se muestran las estructuras de las APIs para el dominio de Servicios de Nómina Corporativa.

Figura 26: Diagrama BOM de Servicios de Nómina Corporativa



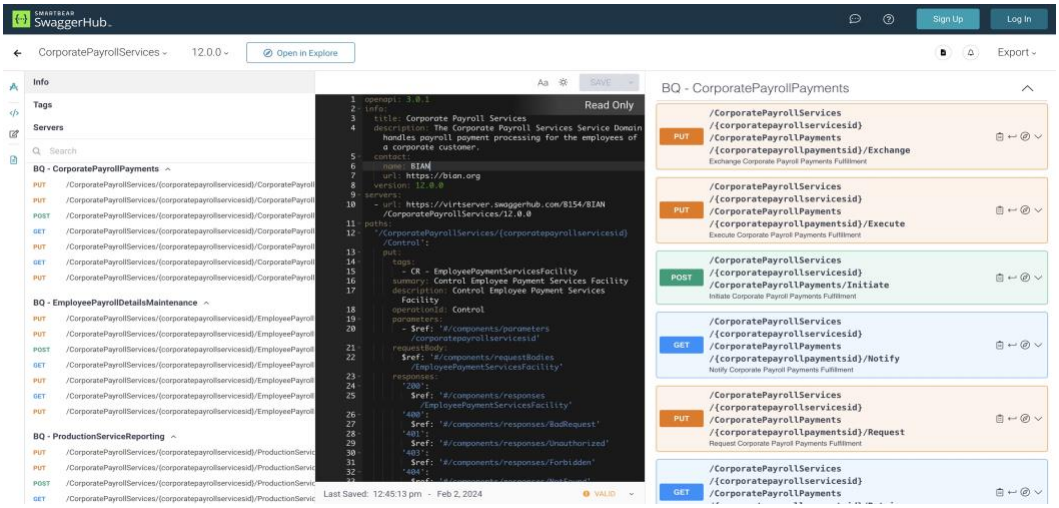
Fuente: BIAN(2025)

Figura 27: Diagrama de registro de control de servicios de nómina corporativa



Fuente: BIAN(2025)

Figura 28: BIAN-Servicios de Nómina Corporativa.



Fuente: BIAN(2025)

4.2 IMPLEMENTACIÓN ARQUITECTURA DE APIS PARA EL PAGO DE NÓMINA EN LÍNEA

4.2.1 DIAGRAMA DE CONTEXTO

El primer paso es realizar un vistazo general de la solución. En la figura 28, "Diagrama de Contexto", se visualizan las interacciones de una nueva empresa afiliada al producto de pago de nómina y sus empleados, con los canales que la entidad financiera expone a todos sus clientes y segmentos con cuentas activas.

La afiliación, suscripción e interacción al canal de pagos de nómina a través de APIs se realiza utilizando los productos de Apigee. El API Manager permite la administración, compilación y protección de las APIs expuestas por la entidad financiera. Estas APIs funcionan como proxys de los microservicios que expone el componente conocido como Procesador de Transacciones.

El Procesador de Transacciones está compuesto por microservicios cuya funcionalidad permite recibir, validar y procesar el pago de las órdenes de pago de nómina que la empresa solicita a la entidad financiera para acreditar a sus empleados en sus respectivas cuentas dentro del Core Bancario. De esta manera,

los empleados de la empresa pueden utilizar los canales de la entidad financiera para acceder a los valores acreditados.

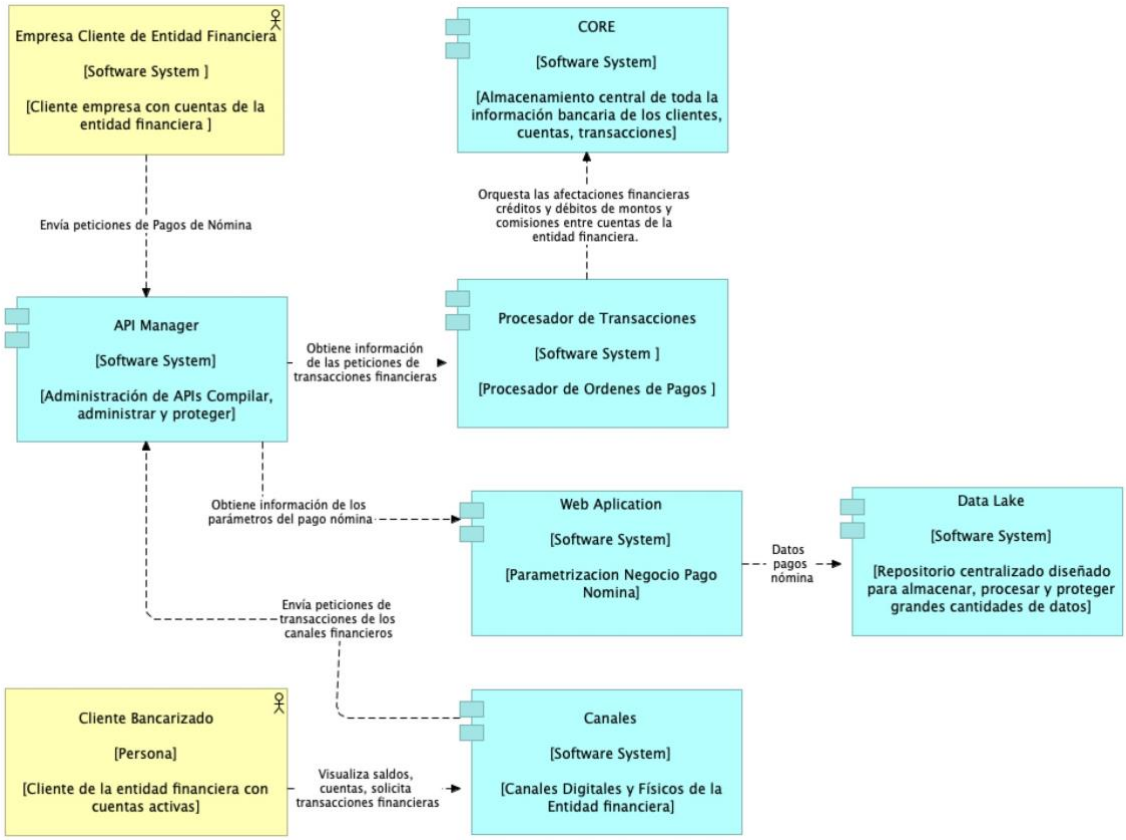
En esta arquitectura, el Procesador de Transacciones es un elemento interno del sistema de información que facilita la integración con los demás sistemas y subsistemas de la entidad financiera. Actualmente, estos sistemas pueden requerir automatización o reducción de deuda técnica como parte de la mejora continua y la entrega de valor. En un contexto de agilidad, el valor para el cliente final se genera únicamente cuando el sistema o producto está operativo.

Como resultado del análisis de la arquitectura empresarial y dentro del marco de mejora continua, se prevé que este componente, en el futuro, evolucione hacia un sistema de información externo especializado en múltiples Cores bancarios.

Los canales, ya sean físicos (como cajeros automáticos [ATM], puntos de venta [POS] y agencias) o digitales (como la web y aplicaciones móviles), permiten a los clientes administrar sus productos financieros, tales como cuentas, tarjetas y préstamos. Gracias a esta integración, la entidad financiera generará ingresos adicionales por la utilización de estos canales, ya sea a través de comisiones o por beneficios derivados de la bancarización.

El Core Bancario registra las afectaciones contables y almacena la información financiera de los clientes, reutilizando funcionalidades existentes, como el cuadre, consolidación y generación de reportes para organismos de control o análisis interno de negocios.

Figura 29: Diagrama de Contexto



Fuente: Propia

4.2.2 DIAGRAMA DE CONTENEDOR

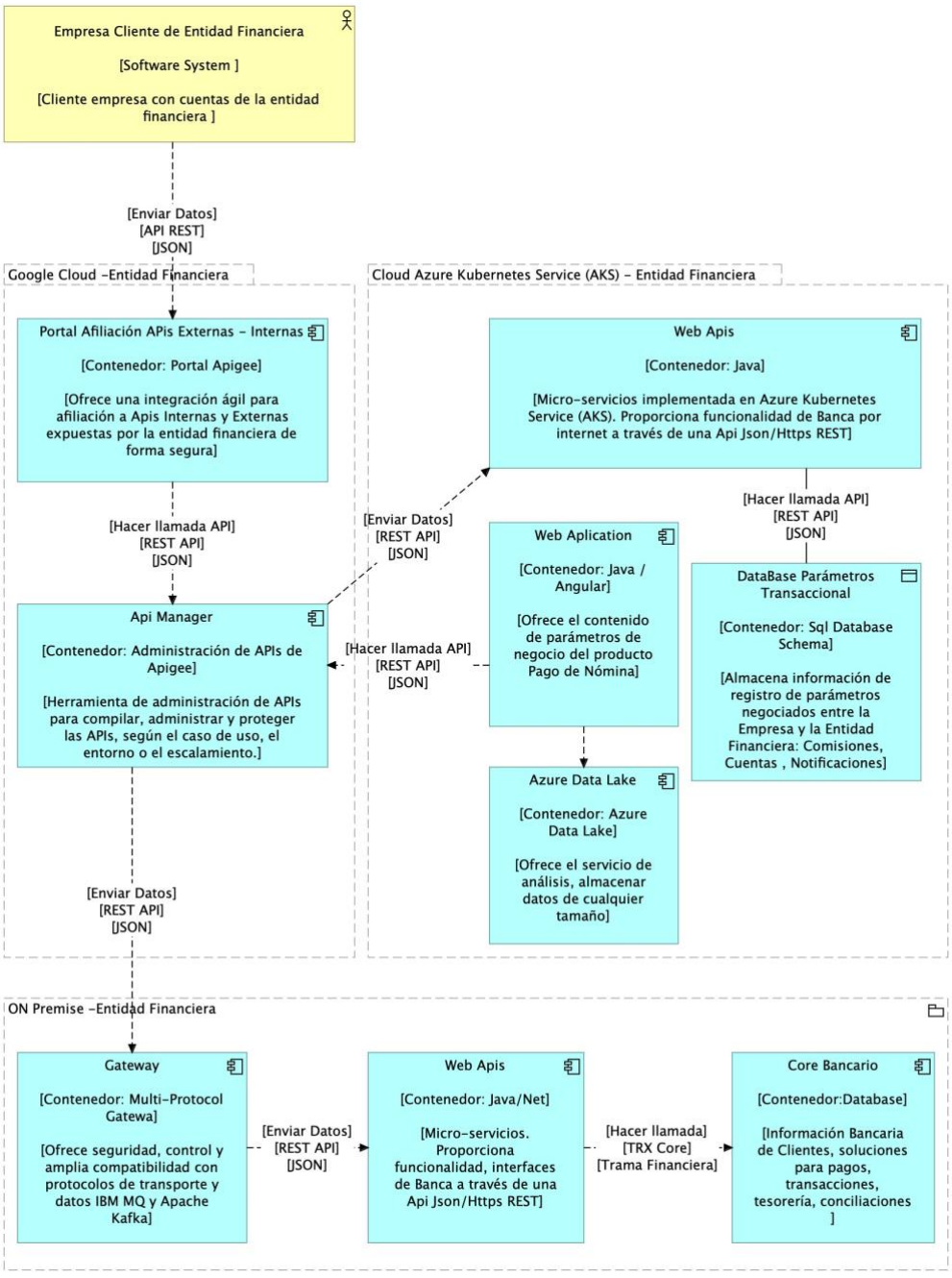
En la figura 29, "Diagrama de Contexto", se visualiza el proceso de afiliación de una empresa al servicio de pago de nómina. La empresa inicia el registro en el portal de Afiliación de APIs externas, donde se encuentra la documentación de las APIs expuestas, tanto de negocio como de seguridad. Además, se detallan los requisitos técnicos necesarios para la integración en diferentes entornos: desarrollo, pruebas y producción.

Para la integración, afiliación y seguridad de las APIs Proxy, se utiliza infraestructura en la nube de Google Cloud. La comunicación entre los componentes se realiza a través de APIs REST con formato JSON.

Los proxys de Apigee gestionan y restringen el acceso a los microservicios backend, los cuales se publican en Azure Kubernetes Services (AKS). Este servicio permite automatizar tareas operativas y administrar contenedores para la implementación de microservicios y del micrositio. Asimismo, los proxys de Apigee controlan el

acceso a la infraestructura on-premise, donde se alojan datos sensibles, como el bus de datos y el Core Bancario.

Figura 30: Diagrama de Contenedor.



Fuente: Propia

4.2.3 DIAGRAMA DE COMPONENTES

El siguiente paso, una vez determinado el alcance de la solución, se centra en la dinámica de comunicación entre los contenedores y la infraestructura existente

destinada a este segmento empresarial. El objetivo es diseñar un ciclo de vida que facilite la comunicación entre la empresa y la entidad financiera para el API de pago de nómina en línea.

En este paso se definen los componentes que se deben implementar y que serán alojados en los contenedores descritos. La figura 30, "Diagrama de Componentes", muestra los elementos según las definiciones de BIAN.

Para la integración de los componentes internos, se aplican los siguientes lineamientos:

Todas las integraciones entre sistemas y subsistemas dentro del ecosistema de la entidad financiera se realizan mediante la exposición y consumo de APIs.

La creación de microservicios seguirá la arquitectura Domain-Driven Design(DDD), con sus tres capas: Aplicación, Dominio e Infraestructura, lo que garantizará la adaptabilidad de los microservicios.

Las integraciones entre microservicios desplegados en la nube privada deberán utilizar un Load Balancer para distribuir el tráfico entrante de pago de nómina entre varios destinos.

La integración con aplicaciones externas bajo el modelo SaaS (Software como Servicio) se realizará a través del API Manager de Apigee.

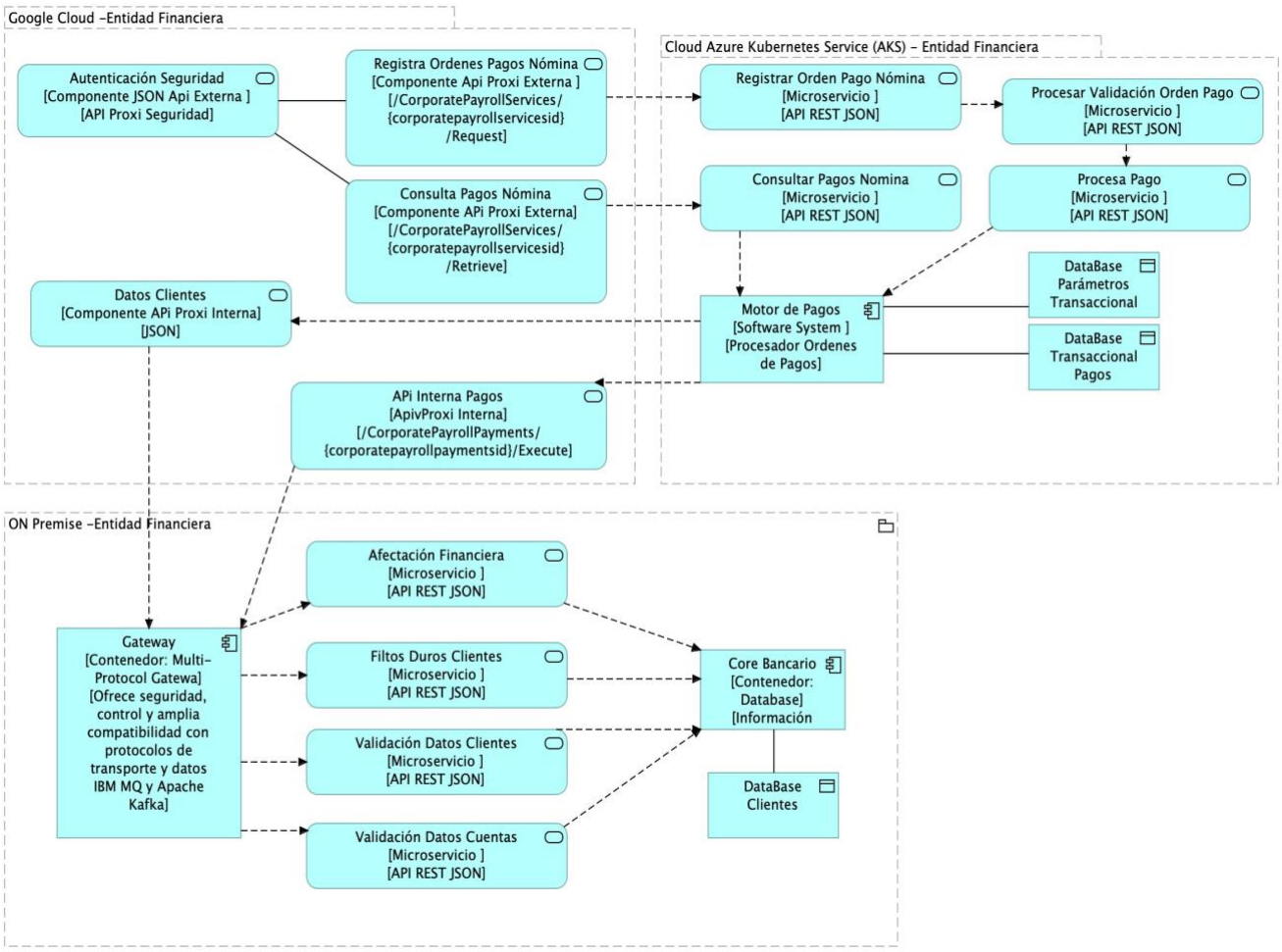
Para garantizar una comunicación segura, se utilizará mTLS v1.2 o versiones superiores. El mecanismo de autorización y autenticación será gestionado por el API Manager.

El componente denominado Procesador de Pagos se basará en una arquitectura de colas, un patrón que utiliza una cola de mensajes para desacoplar las APIs de solicitud de pagos de las APIs transaccionales financieras.

Esta arquitectura permitirá:

- Procesar grandes volúmenes de transacciones de pagos de nómina
- Validar solicitudes de órdenes de pago.
- Detectar y manejar errores en las solicitudes.
- Además, facilitará la gestión de la carga de trabajo y proporcionará escalabilidad al sistema.

Figura 31: Diagrama de Componentes**.**



Fuente: Propia

CONCLUSIONES

LOGROS ALCANZADOS

Según los datos obtenidos, se concluye que la práctica de la arquitectura empresarial es fundamental para identificar oportunidades de negocio, detectar limitaciones técnicas y abordar adecuadamente problemas que podrían afectar la continuidad del negocio. Esto ha resultado en una propuesta de arquitectura de transición que permita sostener a los clientes actuales mientras se afilian nuevas empresas al modelo de negocio B2B para el pago de nómina.

En conclusión, una arquitectura de APIs basada en BIAN, con mensajería Rest, aporta la experiencia y los resultados de grandes líderes en el sector bancario. Nos proporciona una estructura común, sintaxis y modelo de datos para el pago de nómina en el dominio de Servicios de Nómina Corporativa, o cualquier servicio financiero utilizado globalmente, reduciendo la pérdida de información o el manejo de datos irrelevantes que no pertenecen al negocio en cuestión.

De aquí se desprende que, con la estandarización de las APIs, se hace necesario implementar una administración y gobernanza adecuadas, que cubra todo el ciclo de vida de las Apis. En este contexto, el API Manager juega un papel crucial en el diseño de esta arquitectura, permitiendo crear, publicar, controlar, proteger contra ataques, gestionar actualizaciones y facilitar integraciones en el entorno empresarial, incluso con instituciones financieras que tienen infraestructura multi nube. Estas son algunas de las funciones principales que permiten centralizar el control y el análisis de las APIs.

LIMITACIÓN IDENTIFICADA

Con base en los requerimientos identificados para un modelo de negocio B2B, la empresa solicitante del servicio financiero de pago de nómina debe incursionar en la transformación digital para integrarse de manera segura. Por ello, se concluye

que es necesario mantener la solución actual para que las empresas puedan acceder a este servicio financiero, lo que implica que la entidad financiera incurrirá en gastos de mantenimiento de esta plataforma hasta que se priorice la implementación de una nueva plataforma web.

OPORTUNIDADES:

A partir del análisis propuesto en la Fase C: Arquitectura de Sistemas de Información (Figura 18), se identifica una oportunidad de mejora que beneficiará a la entidad financiera, acelerando su crecimiento tecnológico y reduciendo la obsolescencia. Esta oportunidad se refiere a ofrecer el componente identificado como Procesador de Transacciones a un outsourcing de TI. De esta forma, la entidad financiera podría centrarse en su negocio de afiliación de servicios financieros, mientras que el outsourcing se encargaría del procesamiento, incluso llevando el Core Bancario a la nube y/o multi nube.

Con la arquitectura definida, se alcanzan los objetivos planteados en este documento, según los stakeholders identificados en la Fase A: Visión de la Arquitectura (Figura 6). Sin embargo, fuera del alcance de este análisis, se identificó una oportunidad de negocio que requerirá un análisis más detallado, centrado en una empresa de software que ofrezca, gestione, agrupe o administre a aquellas empresas que no tienen la capacidad de establecer una conexión directa con la entidad financiera, actuando como enlace entre ambas partes.

GLOSARIO

Afiliación: Es el acto y el resultado de afiliar. Este verbo hace referencia a la acción de adherir, apuntar, anotar o sumar a un individuo a una asociación u organización (Pérez & Merino, 2022).

SLA: Por sus siglas en inglés "Service Level Agreement" (Acuerdo de Nivel de Servicio), es un contrato firmado entre las partes involucradas en una negociación entre la entidad financiera y la empresa que solicita el servicio.

API: Las API son mecanismos que permiten a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos (AWS, 2023b).

APIGEE: Es una plataforma para desarrollar y administrar APIs. Mediante el fronting de servicios con una capa de proxy, Apigee proporciona una abstracción o fachada para tus APIs de servicio de backend y ofrece seguridad, límite de frecuencia, cuotas, estadísticas y mucho más (Cloud, 2023).

API Manager: Es una herramienta que se utiliza para acceder, distribuir, controlar y analizar las APIs utilizadas por los desarrolladores en un entorno empresarial (IBM, s.f.).

Arquitectura empresarial: Es la disciplina mediante la cual una empresa planifica y organiza sus tecnologías de la información (TI) para alinearlas con su visión empresarial (UNIR, 2023).

Azure: Es una plataforma compuesta por más de 200 productos y servicios en la nube diseñados para ayudar a resolver dificultades actuales y crear el futuro. Permite crear, ejecutar y administrar aplicaciones en varias nubes, en el entorno local y en el perímetro, utilizando las herramientas y marcos preferidos (Azure, s.f.).

Bancarizar: Hacer que alguien o algo, como un grupo social o un país, resuelva sus actividades económicas a través de la banca (Torrente, s.f.).

Estándares Archimate: Es un lenguaje de modelado de arquitectura empresarial abierto e independiente, diseñado para respaldar la descripción, el análisis y la visualización de la arquitectura dentro de los dominios de negocios de manera no ambigua (Hisour, s.f.).

REST: Representational State Transfer es una interfaz para conectar varios sistemas basados en el protocolo HTTP (uno de los protocolos más antiguos), que sirve para obtener y generar datos y operaciones, devolviendo esos datos en formatos específicos como XML y JSON (R. José, 2018).

TOGAF: Son las siglas de The Open Group Architecture Framework (en español, Esquema de Arquitectura del Open Group), un marco de referencia que permite planificar, diseñar e implementar la arquitectura empresarial de una organización (UNIR, 2021).

REFERENCIAS

BIAN(2025) the Banking Industry Architecture Network. Obtenido de https://bian.org/

AWS. (2023). ¿Qué es una interfaz de programación de aplicaciones (API)? *Amazon Web Services* (AWS). Recuperado https://aws.amazon.com/es/what-is/api/

Red Hat (2024). ¿Qué son y para qué sirven los microservicios? Beneficios de los microservicios.Recuperado de https://www.redhat.com/es/topics/microservice.

AKS. (s.f.). Compilación de una canalización de CI/CD para microservicios en Kubernetes con Azure DevOps y Helm. Recuperado de https://learn.microsoft.com/es-es/azure/architecture/microservices/ci-cd-kubernetes.

Microsoft Azure. (s.f.). Qué es Azure. Obtenido de https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-azure#:~:text=Azure%20Managed%20Lustre%20es%20un,(HPC)%20en%20la%20nube.

Microsoft Azure. (s.f.). CI/CD para arquitecturas de microservicios. Recuperador https://learn.microsoft.com/es-es/azure/architecture/microservices/ci-cd.

Begum, M., & Raaj, V. (2021). Analysis and adoption of mobile banking: An e-banking system. En 7th International Conference on Advanced Computing and Communication Systems (ICACCS), marzo de 2021. ISSN 2575-7288. Recuperado de https://bibliotecas.ups.edu.ec:2095/document/9441778

CEUPE. (s.f.). ¿Qué es una empresa? Centro Europeo de Postgrado (CEUPE). Recuperado de https://www.ceupe.com/blog/gran-empresa.html

Cloud, G. (2023). Qué es Apigee. Google Cloud. Recuperado de https://cloud.google.com/apigee/docs/api-platform/get-started/what-apigee-old?hl=es-419

Coppolino, L., D'Antonio, S., Romano, L., Campanile, F., & de Carvalho, A. V. (2015). Effective visualization of a big data banking application. En E. Damiani, R. J. Howlett, L. C. Jain, L. Gallo, & G. De Pietro (Eds.), Intelligent Interactive Multimedia Systems and Services (pp. 57–68). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-19830-9

DevOpsl. (s.f.). Diseño de arquitectura de microservicios. Emiral FG. Recuperado de https://emiralfg.com/que-es-erp-que-significa-para-que-sirve/

DevOpsl. (2023). Uso de análisis de dominios para modelar microservicios. Microsoft Learn. Recuperado de https://learn.microsoft.com/es-es/azure/architecture/microservices/model/domain-analysis

Evelin, H., & Norma, S. (2013). Desarrollo de un sistema web y móvil para gestionar los procesos de reservación y pagos en línea del hotel Miravalle del cantón Mejía aplicando la metodología Kanban. *Trabajo de grado.*

Gualli, F. (2019). Desarrollo de un sistema web de prototipo para facturación electrónica de una ferretería. Trabajo de grado. Recuperado de https://repositorio.unan.edu.ni/9719/1/19031.pdf

Hisour. (s.f.). ArchiMate. Hisour. Recuperado de https://www.hisour.com/es/archimate-27816/

IBM. (s.f.). Qué es API Management. IBM. Recuperado de https://www.ibm.com/es-es/topics/api-management

Bodega, J. (2019a). Aplicación de metodologías de creación de APIs en diferentes frameworks de desarrollo web utilizando elementos ORM. Trabajo de grado. Universidad Politécnica de Madrid. Recuperado de https://oa.upm.es/55691/1/TFG_JORGE_BODEGA_FERNANZ.pdf

Bonilla, J. (2019b). Desarrollo de un sistema para gestión de roles de pago con el envío de notificaciones en tiempo real. Trabajo de grado. Escuela Politécnica Nacional. Recuperado de https://bibdigital.epn.edu.ec/bitstream/15000/20383/1/CD9855.pdf

Valladarez, J. (2020). ¿Qué son APIs? Universidad Panamericana. Recuperado de https://ingenieria.upana.edu.gt/wp-content/uploads/2020/07/Edutek-dia-5-RESTful-api-RESTCONF2.pdf

Josey, A. (2013). TOGAF® Versión 9.1 - Guía de Bolsillo. Van Haren. ISBN 978-90-8753-710-4. Recuperado de https://books.google.com.ec/books?id=6elEBAAAQBAJ

José, R. (2018). ¿Qué es REST: Conoce su potencia? *OpenWebinars*. Recuperado de https://openwebinars.net/blog/que-es-rest-conoce-su-potencia/

Laura, B., & Mariela, A. (2022). Nómina 360 y procesos de pago compañía Plantabal S.A. Ecuador. *Trabajo de grado*

Mero-Jessica. (2018). Empresa, administración y proceso administrativo. Trabajo de grado. Ecuador.

Migratel, A. (s.f.). Uso de un diseño basado en dominios táctico para diseñar microservicios. Microsoft Learn. Recuperado de https://learn.microsoft.com/es-es/azure/architecture/microservices/model/tactical-ddd

Monitor, A. (s.f.). Supervisión de una aplicación de microservicios en AKS. Microsoft Learn. Recuperado de https://learn.microsoft.com/es-es/azure/architecture/microservices/logging-monitoring

Nosrati, L., & Bidgoli, A. M. (2015). Security assessment of mobile-banking. En 2015 International Conference and Workshop on Computing and Communication (IEMCON). Recuperado de https://bibliotecas.ups.edu.ec:2095/document/7344489

Pathak, R., & Joshi, S. (2009). SMC protocol for privacy preserving in banking computations along with security analysis. En 2009 First Asian Himalayas International Conference on Internet (pp. 1–5). https://doi.org/10.1109/AHICI.2009.5340328

Primicias. (s.f.). Pymes ecuatorianas: Un vistazo en datos. Primicias. Recuperado de https://www.primicias.ec/nota_comercial/hablemos-de/empresas/actualidad-empresas/pymes-ecuatorianas-transformacion-crecimiento-exponencial/

Pérez, J., & Merino, M. (2022). Definición de afiliación. Definicion.de. Recuperado de https://definicion.de/afiliacion/

Ruiz, F. (2020). APIs:¿Qué son y cómo están transformando el sector financiero? Finero Connect Blog. Recuperado de https://blog.finerioconnect.com/apis-que-son-y-como-estan-transformando-el-sector-financiero/

Sandoval, F., Galvez, V., & Moscoso, O. (2017). Desarrollo de arquitectura empresarial usando un framework con enfoque ágil. Enfoque UTE, 8(1), 135–147. Recuperado de https://www.redalyc.org/journal/5722/572262176010/html/

Summit, B. (s.f.-a). Acerca de Bian. BIAN. Recuperado de https://bian.org/members/faq/about-bian-2/

Summit, B. (s.f.-b). BIAN Service Landscape V12.0 Matrix View. BIAN. Recuperado de https://bian.org/servicelandscape-12-0-0/views/view_51891.html

Torrente. (s.f.). Bancarizar, neologismo del lenguaje económico. Fundéu. Recuperado de https://www.fundeu.es/recomendacion/bancarizar/

UNIR. (2023). ¿Qué es la arquitectura empresarial y para qué sirve? Universidad Internacional de La Rioja (UNIR). Recuperado de https://www.unir.net/empresa/revista/arquitectura-empresarial/

UNIR. (2021). ¿Qué es la metodología Togaf? Universidad Internacional de La Rioja (UNIR). Recuperado de https://www.unir.net/empresa/revista/togaf-que-es/

Yungrid, E. V., & Carlos, R. (2012). Revisión de la definición de pyme en América Latina. Panamá.

AWS. (2023b). Qué es una interfaz de programación de aplicaciones (API). Amazon Web Services (AWS). Recuperado de https://aws.amazon.com/es/what-is/api/