



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Sistemas

**Un framework como solución para mejorar la gestión
de servicios REST requeridos en los pases a
producción de una entidad bancaria**

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de Ingeniera de Sistemas

AUTOR

Carolay Piedad DIAZ GOZZER

ASESOR

Luis Alberto ALARCÓN LOAYZA

Lima, Perú

2021



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Díaz, C. (2021). *Un framework como solución para mejorar la gestión de servicios REST requeridos en los pases a producción de una entidad bancaria*. [Trabajo de suficiencia profesional de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Sistemas]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	CAROLAY PIEDAD DIAZ GOZZER
DNI	42500230
URL de ORCID	-
Datos de asesor	
Nombres y apellidos	LUIS ALBERTO ALARCON LOAYZA
DNI	00456684
URL de ORCID	https://orcid.org/0000-0001-8340-7161
Datos de investigación	
Línea de investigación	Desarrollo de modelos y aplicación de las tecnologías de información y comunicaciones.
Grupo de investigación	“ITDATA”
Agencia de financiamiento	-
Ubicación geográfica de la investigación	Perú, Lima, Lima, Pueblo Libre, Parque San Martín 313, departamento 103. Latitud: -12.070470 Longitud: -77.064163
Año o rango de años en que se realizó la investigación	2021
URL de disciplinas OCDE	Ingeniería de Sistemas y Comunicaciones https://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
Escuela Profesional de Ingeniería de Sistemas

**Acta Virtual de Sustentación
del Trabajo de Suficiencia Profesional**

Siendo las 20:00 horas del día 06 de agosto del año 2021, se reunieron virtualmente los docentes designados como Miembros de Jurado del Trabajo de Suficiencia Profesional, presidido por el Mg. Moquillaza Henriquez Santiago (Presidente), Lic. Contreras Flores Walter Pedro (Miembro) y el Lic. Alarcón Loayza, Luis Alberto (Miembro Asesor), usando la plataforma Meet (<https://meet.google.com/hby-rhye-eji>), para la sustentación virtual del Trabajo de Suficiencia Profesional intitulado: **“UN FRAMEWORK COMO SOLUCIÓN PARA MEJORAR LA GESTIÓN DE SERVICIOS REST REQUERIDOS EN LOS PASES A PRODUCCIÓN DE UNA ENTIDAD BANCARIA”**, por la Bachiller **Díaz Gozzer Carolay Piedad**; para obtener el Título Profesional de Ingeniera de Sistemas.

Acto seguido de la exposición del Trabajo de Suficiencia Profesional, el Presidente invitó a la Bachiller a dar las respuestas a las preguntas establecidas por los miembros del Jurado.

La Bachiller en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las observaciones y preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, la Bachiller obtuvo la nota de 16 Dieciséis.

A continuación el Presidente de Jurados el Mg. Moquillaza Henriquez Santiago, declara a la Bachiller **Ingeniera de Sistemas**.

Siendo las 21:24 horas, se levantó la sesión.

Presidente

Mg. Moquillaza Henriquez Santiago

Miembro

Lic. Contreras Flores Walter Pedro

Miembro Asesor

Lic. Alarcón Loayza, Luis Alberto

Ficha Catalográfica

Un FRAMEWORK como solución para mejorar la Gestión de Servicios REST requeridos en los pases a producción de una entidad bancaria.

AUTORA: DIAZ GOZZER CAROLAY PIEDAD

ASESOR: ALARCÓN LAOYZA LUIS ALBERTO

LIMA - PERÚ, 2021

Título Profesional/Grado Académico: Título Profesional de Ingeniero de Sistemas.

Área / Programa / Línea de Investigación: Ingeniería / Tecnología de la Información y Comunicación / Ingeniería de Software.

Pregrado: Universidad Nacional Mayor de San Marcos - Facultad de Ingeniería de Sistemas e Informática - Escuela Profesional de Ingeniería de Sistemas.

Formato 28 x 20 cm

Páginas: 85

Referencia bibliográfica

Díaz, C. (2021). Un FRAMEWORK como solución para mejorar la Gestión de Servicios REST requeridos en los pases a producción de una entidad bancaria. Tesis para optar el título profesional de Ingeniero de Sistemas. Escuela Profesional de Ingeniería de Sistemas, Facultad de Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima, Perú.

Dedicatoria

A mis padres María Gozzer de Díaz y Sanoé Díaz Deza,
quienes con mucho amor y disciplina me apoyaron en todas mis decisiones.

Por permitirme fallar y alentarme a continuar con más ímpetu y fuerzas.

Agradecimientos

Agradezco a Dios y a mi gran familia, mis padres, hermanos y sobrinos que con gran amor me alentaron a seguir adelante.

Agradezco a todas aquellas personas que me brindaron su tiempo, energías y dedicación para poder lograr cada una de mis metas.

Agradezco a mis compañeros y colegas quienes me enseñaron a trabajar en grupo y confiar en el equipo.

Agradezco a mi asesor por su tiempo, dedicación y por trasmitirme de su valiosa experiencia.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Un FRAMEWORK como solución para mejorar la Gestión de Servicios REST requeridos en los pases a producción de una entidad bancaria

Autor: Díaz Gozzer Carolay Piedad.
Asesor: Alarcón Loayza, Luis Alberto.
Título: Trabajo de Suficiencia Profesional para optar el Título Profesional de Ingeniero de Sistemas.
Fecha: Julio de 2021.

Resumen

En el presente trabajo de suficiencia profesional se describe la implementación de un *framework* para mejorar la gestión de Servicios Web REST que brinda una empresa de Outsourcing de TI a una entidad bancaria. Es importante recalcar que dicho *framework* fue planteado como solución a un problema, el cual se veía reflejado en los pases a producción fallidos con significativas demoras en la atención y en la falta de conocimiento de las funciones o roles requeridos en el desempeño de una de las áreas más importantes de ingeniería de uno de los más importantes bancos del Perú.

Como marco metodológico se usó ISTQB Agile Testing y otros enfoques ágiles, logrando generar una solución efectiva considerando todos los flujos y roles para el proceso de mejora continua en el ciclo de vida de los Servicios Web REST.

Como resultado esperado contamos, hoy en día, con una solución de gestión de servicios REST que permita optimizar los proyectos de desarrollo en una entidad bancaria.

Palabras Claves: Framework, Servicios REST, Outsourcing, ISTQB Agile Testing.

NATIONAL UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEMS ENGINEERING
PROFESSIONAL SCHOOL OF SYSTEMS ENGINEERING

A FRAMEWORK as a solution to improve the Management of REST Services required in the pass to production of a bank

Author: Diaz Gozzer Carolay Piedad.

Adviser: Alarcon Loayza, Luis Alberto.

Title: Work of Professional Sufficiency to opt for the Professional Title of Systems Engineer.

Date: July 2021.

Abstract

This professional sufficiency project describes the implementation of the framework in order to improve REST service management for banking IT outsourcing. It's important to emphasize that this framework was designed as a solution based in failed passes processes delays, lack of productivity, required positions for specific activities in engineering areas in one of the most important banks in Peru.

The methodology used for this project is ISTQB Agile Testing and others agile frameworks, were each an effectiveness solution considering all the flows and roles for the continuous improvement process in the lifecycle of REST Web Services.

Our main purpose is to have a REST service management that will allow to optimize development projects of the entity.

Keywords: Framework, REST Service, Outsourcing, ISTQB Agile Testing.

Tabla de contenido

Acta de sustentación	3
Ficha Catalográfica	4
Referencia bibliográfica	5
Dedicatoria	6
Agradecimientos	7
Resumen	8
Abstract	9
Introducción	16
Capítulo I Trayectoria profesional	18
Presentación Profesional	18
Experiencia Laboral	18
Formación Académica	20
Certificados	20
Idiomas	21
Capítulo II Contexto en el que se desarrolló la experiencia	22
2.1 Empresa - Actividad que realiza	22
2.2 Visión	24
2.3 Misión	25
2.4 Organización de la empresa	25
2.5 Área, cargos y funciones desempeñadas	26
2.6 Experiencia profesional ejecutada en la organización	28
Capítulo III Actividades desarrolladas	29
3.1 Situación Problemática	36

3.1.1	Definición del problema	37
	Problemas Específicos.....	37
3.2	Solución	38
3.2.1	Objetivos	38
3.2.2	Alcance	39
3.2.3	Etapas y Metodologías	41
3.2.3.1	Etapas:.....	41
3.2.3.2	Metodologías:	51
3.2.4	Fundamentos Utilizados.....	51
3.2.4.1	ISTQB Agile Testing	51
3.2.4.2	SCRUM.....	54
3.2.4.3	KANBAN	55
3.2.4.4	ITIL v.4	57
3.2.5	Implementación de la solución	57
3.3	Evaluación.....	68
	Capítulo IV Reflexión Crítica de la Experiencia	72
	Capítulo V Conclusiones y Recomendaciones	73
1.1	Conclusiones	73
1.2	Recomendaciones:	74
	Bibliografía	75
	Anexos	77

Índice de Tablas

Tabla 1: Experiencia laboral	18
Tabla 2: Experiencia laboral (continuación).....	19
Tabla 3: Formación académica	20
Tabla 4: Certificados	20
Tabla 5: Idiomas	21
Tabla 6: Operaciones RESTful.	31
Tabla 7 : Fases por Tiempo de Atención.	48
Tabla 8: Herramientas de soporte al framework.....	53
Tabla 9: Flujo del Proceso de Diseño.	57
Tabla 10: Flujo del Proceso de Solicitud de Información.	58
Tabla 11: Flujo del proceso de Construcción.	59
Tabla 12: Flujo del Proceso de Configuración.	60
Tabla 13: Flujo del Proceso de Test de Arquitectura.....	61
Tabla 14: Flujo del Proceso de Test Aplicativo.....	62
Tabla 15: Flujo del Proceso de Calidad Aplicativo.	63
Tabla 16: Flujo del Proceso de Calidad Aplicativo (continuación).....	63
Tabla 17: Flujo del Proceso de Pase a Producción.	65
Tabla 18: Flujo del Proceso de Pase a Producción.	66
Tabla 19: Flujo del Proceso de Post Pase a Producción.	67
Tabla 21: Consideraciones para el Reuso de Servicios.	77
Tabla 22: Consideraciones para el Reuso de Servicios (continuación)	78
Tabla 23: Consideraciones para Modificación de Servicios REST.....	79
Tabla 24: Consideraciones para Modificación de Servicios REST (continuación).....	80

Tabla 25: Consideraciones para la creación de Servicios REST.....	81
Tabla 26: Consideraciones para la creación de Servicios REST (continuación).....	82

Índice de Figuras

Figura 1: Indicadores ENTELGY en el mundo	23
Figura 2: Divisiones de Entelgy.....	24
Figura 3: Valores Entelgy	25
Figura 4: Organigrama Entelgy Perú	26
Figura 5: Comunicación Cliente - Servidor de Servicios REST.....	30
Figura 6: Operaciones con método GET.	30
Figura 7: Canales de distribución.	32
Figura 8: Ciclo de vida de Servicios REST.	34
Figura 9: Arquitectura de servicios.....	35
Figura 10: Problema y soluciones propuestas.....	38
Figura 11: Flujo General del Ciclo de Vida de Servicios REST.	40
Figura 12: Diagrama Fases del Framework.	41
Figura 13: Modelo de T907.	49
Figura 14: Modelo de P110 - Hoja de Pase	50
Figura 15: Ether Catalogs.	52
Figura 16: Ether Catalogs (otra vista).....	53
Figura 17: Ejemplo de especificación de Casos de Prueba.....	54
Figura 18: Seguimiento Retrospectiva de Post Pase a Producción.	55
Figura 19: Jira de Arquitectura de Servicios REST.....	56
Figura 20: Especificaciones de un ticket.	56
Figura 21 : Pases a Producción – Periodo: enero 2019 a junio 2019.....	68
Figura 22: Pases a Producción – periodo enero 2020 a junio 2020.	70
Figura 23: Motivos de Rollback – periodo enero 2020 a junio 2020.	71

Figura 24: <i>Checklist</i> de Gestión para Pase a Producción de Servicios.	83
Figura 25: Flujo de Atención de Fallas en Ambientes Previos.....	84
Figura 26: Flujo de Pases a Producción y ejecución de cambios en caliente.	85
Figura 27: Flujo de Pruebas Integrales de Servicios REST.....	86

Introducción

El actual trabajo tiene como finalidad presentar un *framework* que se implementó como solución para la mejora de la gestión de pases a producción de servicios REST. ENTELGY (como empresa prestadora de servicios) brinda este servicio a uno de los más importantes bancos de Perú.

El nivel de servicio que se hará referencia encierra diferentes aspectos como son la definición de roles, funciones y tiempos de atención, el aseguramiento en los flujos de procesos y en la disminución en los índices de pases a producción fallidos. De este modo, se presentará la estructura, implementación y ajuste de este marco de trabajo.

El presente trabajo abarca los siguientes capítulos:

En el Capítulo I tiene como objetivo detallar de manera cronológica la trayectoria del autor. Se especifica las funciones desarrolladas en cada experiencia y su desarrollo académico, tal como son, estudios extracurriculares, cursos de especialización, certificaciones internacionales e idiomas.

En el Capítulo II se tocan aspectos relacionados a la empresa como son la visión y misión, organización, valores y algunos indicadores sobre ENTELGY en el mundo. Además, se detalla el aporte profesional que ha brindado la autora a la empresa en la que se ha desarrollado el presente trabajo de suficiencia profesional.

En el Capítulo III abarca la parte principal de este trabajo, es decir, el desarrollo de la solución. Se describe el análisis que se realizó, las etapas de análisis situacional, la definición de roles y funciones, ejecuciones de tareas por tiempo de atención; y, por último, el refinamiento.

En el Capítulo IV se menciona todos aquellos aspectos no relacionados a la implementación de la solución, sino más bien al aporte de la autora con respeto a la solución planteada. Este aporte se detalla de manera crítica y concisa.

Finalmente, en el Capítulo V se describirán las conclusiones a las que ha llegado la autora en cuanto a su experiencia y a las recomendaciones que haría para poder proponer alguna mejora que no se atendió en la solución.

Capítulo I

Trayectoria profesional

Presentación Profesional

La autora del presente trabajo cuenta con el grado de Bachiller en la carrera de Ingeniería de Sistemas de la Universidad Nacional Mayor de San Marcos. Cuenta con cuatro años de experiencia profesional de los cuales tres años las realizó como egresada de la universidad.

Además, ha podido desarrollar sus conocimientos y ha adoptado muchas habilidades que le han abierto paso en esta línea de carrera. La autora se considera una persona activa, puntual, con una gran capacidad de análisis, ordenada y sistemática, con gran interés por el aprendizaje continuo que conlleve al desarrollo del área en la que se desempeña.

Experiencia Laboral

A continuación, detallo cada una de mis experiencias laborales empezando por mi ocupación actual, indicando: el periodo, cargo y funciones desempeñadas.

Tabla 1: Experiencia laboral
Fuente: Elaboración propia

Institución y Cargo	Funciones	Periodo
ENELGY SAC Cargo: Analista de Incidencias de Nivel 1 de Mantenimiento de Aplicaciones en BBVA	Analista encargada del diagnóstico y seguimiento de Incidencias de primer nivel. Aseguramiento del cumplimiento de tiempos de atención SLA. Aseguramiento del flujo de atención. Actualización de la matriz de escalamiento y Gestión de Conocimiento. Derivación de incidencias a diversos grupos resolutores proporcionando información recogida del monitoreo de servicios y documentación técnica a nivel de la capa de servicios ASO y <i>Backend</i> del negocio.	10/2020 a la fecha
ENELGY SAC Cargo: Analista de Aseguramiento de la Calidad y Gestión de Servicios en BBVA	Gestión de Pases a Pre-producción y Producción. Soporte a ambientes previos y productivos. Planteamiento de Pruebas y Ejecución de estas. Ejecución de Pruebas a Servicios y detección de errores. Aseguramiento de Calidad, definición y afinación de	01/2019 a 09/2020

los diferentes procesos y entregables.

Tabla 2: Experiencia laboral (continuación)
Fuente: Elaboración propia

Institución y Cargo	Funciones	Periodo
EQUIFAX Cargo: Analista de Calidad de Software	Planteamiento de la Estrategia de Calidad y Plan de Pruebas. Definición del Cuadrante de Pruebas Ágiles. Aseguramiento de la Metodología Scrum (ágil) Análisis y planteamiento de pruebas de calidad. Ejecución de pruebas funcionales y no funcionales. Implementación y despliegue en los diferentes ambientes (Calidad y Producción). Gestión de Pases a Producción. Desarrollo y Certificación sobre Scaled Agile Framework SAFe.	07/2017 a 11/2018
UNMSM – QUIPUCAMAYOC Cargo: Técnico en Arquitectura de Software.	Instalación y configuración de entornos de prueba y desarrollo Jenkins. Gestión y administración de los Servidores de Aplicaciones TomCat y JBoss. Gestión de Accesos a la Base de Datos Oracle. Apoyo en tareas de Administración de Base de Datos Oracle como Monitoreo de Carga del Servidor, Backups, Refinamiento, etc. Aseguramiento de la Calidad de Código con SonarQube. Realizar y apoyar la toma de decisiones respecto a la selección de tecnologías. Crear Manuales de Instalación, configuración, documentación de Procesos.	03/2016 a 07/2017
NEGINTEC Practicante Externo de Consultoría SAP BI – AUNA	Elaboración de Documentación Técnica. Programación ABAP. Creación de Querys	01/2014 a 11/2014

Formación Académica

Tabla 3: Formación académica
Fuente: Elaboración propia

Centro de Estudios	Grado Académico	Periodo
Universidad Nacional Mayor de San Marcos Facultad de Ingeniería de Sistemas e Informática.	Bachiller en Ingeniería de Sistemas de la Escuela Académica Profesional en Ingeniería de Sistemas.	Marzo 2011 - Julio 2017
Instituto Peruano de Marketing Carrera Profesional de Marketing	Profesional en Marketing	Enero 2009 - diciembre 2009
Instituto Peruano de Arte y Diseño Carrera Profesional de Diseño Publicitario	Profesional en Diseño Publicitario	Marzo 2004 - diciembre 2007
Colegio Integral Bellavista Campamento Minero Cerro de Pasco Educación Secundaria	Educación Secundaria	Marzo 1996 - diciembre 2002

Certificados

Tabla 4: Certificados
Fuente: Elaboración propia

Certificado	Código	Fecha
Scrum ProductOwnerCertified Expert – SPOCE	xUCXzR6JTy	Agosto 2020
Scrum Master Certified Expert – SMCE	2Wo36FrGOX	Agosto 2020
Scrum DeveloperCertified Expert – SDCE	Rrpz5olXGH	Agosto 2020
Scrum FoundationsCertified Expert - SFCE	qJ412Eo9js	Agosto 2020
Certificación ISTQB Agile Testing	924143	Febrero 2020
Certificación ISTQB FoundationLevel	63412	Febrero 2019

Idiomas

Tabla 5: Idiomas
Fuente: Elaboración propia

Idioma	Nivel
Inglés Americano	Intermedio
Italiano	Básico
Portugués	Básico

Capítulo II

Contexto en el que se desarrolló la experiencia

Es importante precisar que la autora del presente trabajo de suficiencia profesional se desempeñó en la modalidad *bodyshopping*, es decir, siendo un personal externo (proveedor) desarrolló sus actividades como si fuera parte de la organización del cliente. Sin embargo, la descripción de los siguientes apartados estará enfocado a la empresa proveedora del servicio.

2.1 Empresa - Actividad que realiza

ENTELGY es una empresa de origen español que brinda servicios tecnológicos en la modalidad de outsourcing como:

- Servicios de Rendimiento de TI como disciplina basada en la gestión de procesos.
- Servicio de administración remota para garantizar la disponibilidad y rendimiento de las infraestructuras de nuestros clientes.
- Servicio de fábrica de software para el desarrollo de software y la generación de valor: Agile y Software Development.
- Transformación Ágil para la gestión e implantación de agilidad para contribuir a la construcción de productos y servicios de valor.
- Auditoría Interna de Aplicaciones, basándose en estándares del mercado y “buenas prácticas”, identifica los puntos de mejora y de evolución de las operaciones, tanto en la implementación de las aplicaciones como en los procesos asociados a dicha implementación.
- Servicios integrales de implementación, soporte y mantenimiento sobre las siguientes aplicaciones de Oracle PeopleSoft: ERP Finanzas y Cadena de Suministro; HCM y Nómina; y Campus Solutions (Académico).

- Solución de Firma Digital de Entelgy que facilita a las organizaciones obtener las evidencias de una operación de firma remota de una manera segura.

Tiene participación en 8 países entre los cuales se encuentran Argentina, Brasil, Chile, Colombia, España, México, Perú y Estados Unidos de América. (ENTELGY, 2021)

ENTELGY se considera a sí misma como un acelerador global de la transformación para quienes necesitan seguir siendo competitivos en un mundo cambiante a gran velocidad. Definición que hoy en día toma más importancia en un mundo que está pasando por una devastadora pandemia, la misma que ha incentivado (u obligado) a muchas empresas a optar por el cambio digital.(ENTELGY, 2021)



Figura 1: Indicadores ENTELGY en el mundo
Fuente:(ENTELGY, 2021)

ENTELGY trabaja para aportar el máximo valor a sus clientes en el descubrimiento, adopción, consolidación y mantenimiento de nuevas tecnologías que impulsen su negocio, siempre *ciberprotegido*. (ENTELGY, 2021)

Para ofrecer soluciones completas e integradas, en ENTELGY trabajamos en 4 divisiones:

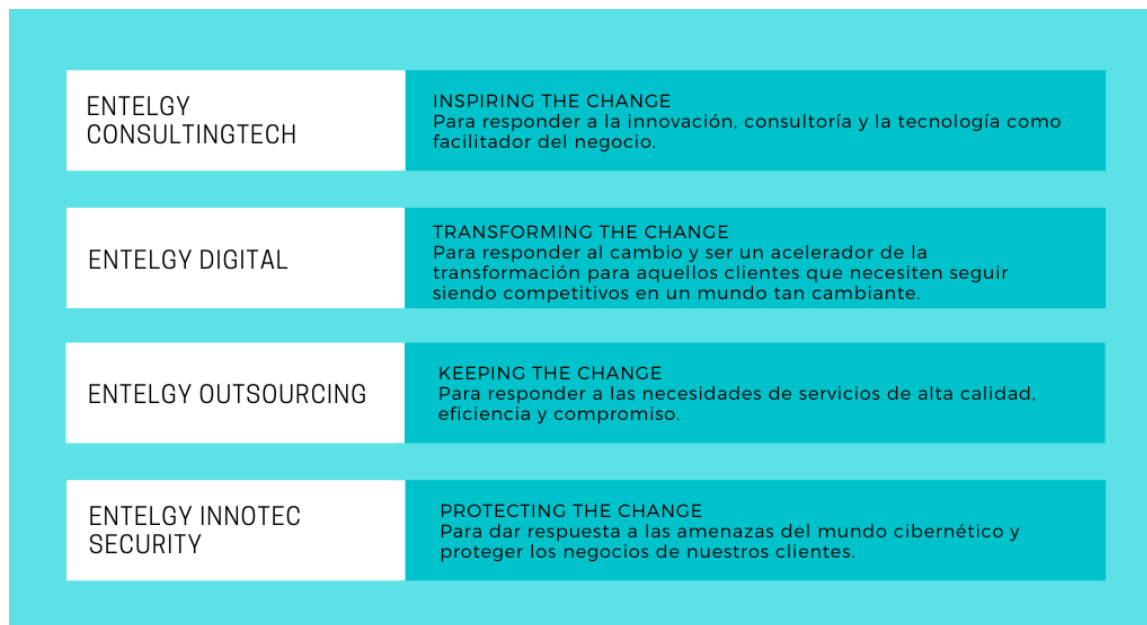


Figura 2: Divisiones de Entelgy
Fuente: (ENTE LGY, 2021)

2.2 Visión

ENTE LGY define su visión global como la de “aportar más valor mediante la innovación”. Ser pieza clave en el mundo del cambio digital y continuando con el propósito de contribuir a un mundo mejor, aportando cada vez mayor valor tecnológico y creando más riqueza, bienestar y empleo, mediante el desarrollo de un proyecto empresarial rentable, sólido, sostenido e innovador, y asimismo más apasionante y feliz, en la que todos y cada uno de sus colaboradores se sienta muy orgulloso de formar parte.

2.3 Misión

ENELGY centra su misión en trabajar día a día con la finalidad de contar con profesionales que sean:



Figura 3: Valores Entelgy
Fuente: (ENELGY, 2021)

2.4 Organización de la empresa

ENELGY actualmente cuenta con más de 1.900 profesionales repartidos en 8 países los cuales brindan diferentes servicios a un total de 300 clientes alrededor del mundo.

ENTELGY tiene 8 años en el Perú y actualmente tiene en planilla a más de 130 profesionales de diferentes lugares del Perú trabajando actualmente en remoto.

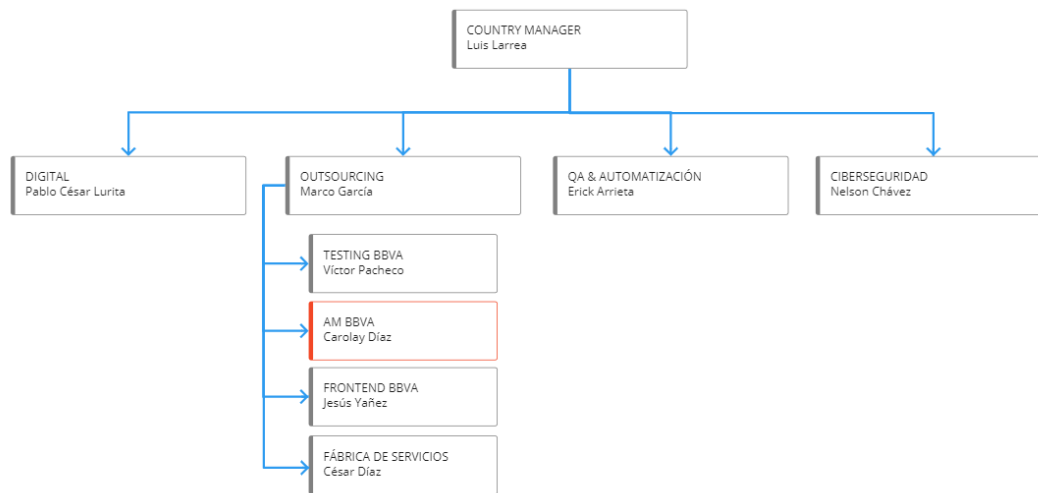


Figura 4: Organigrama Entelgy Perú
Fuente: Elaboración propia

2.5 Área, cargos y funciones desempeñadas

En el periodo en el que se desarrolló el proyecto que detalla el presentetrabajo de suficiencia profesional, la autora se desempeñó en el rol de Gestora de Servicios y Analista del Aseguramiento de la Calidad en el Área de Arquitectura de Servicios de la entidad bancaria desde enero del 2019 hasta octubre del 2020 en la modalidad de *bodyshopping* (personal externo).

Las funciones que desempeñó en esta área fueron las siguientes:

- Revisión del cumplimiento de los componentes según las especificaciones técnicas de los servicios REST.
- Diseño y planteamiento de los diferentes procesos de revisión, implementación, integración, despliegue y atención de incidencias productivas.
- Ejecución de *checklist* para pases productivos.
- Análisis, réplica y diagnóstico de incidencias productivas.

- e. Derivar incidencias a diferentes grupos resolutores.
- f. Preparación de capacitaciones a nuevo personal para la integración en equipos Scrum.
- g. Aseguramiento del cumplimiento de *checklist* y gestión de servicios a gestores subalternos.
- h. Ejecución de pruebas y documentación de evidencias (T907)
- i. Realizar despliegues en los diferentes entornos pre productivos y productivos.
- j. Atención, análisis y documentación de incidencias presentadas en la fase de pruebas.
- k. Soporte técnico a los diferentes Equipos SCRUM.
- l. Aseguramiento del cumplimiento de *checklist*.
- m. Aseguramiento del proceso, cumplimiento de tiempos de atención y otros puntos de acuerdo con los ANS.
- n. Coordinación de Pruebas de No Impacto.
- o. Coordinación y preparación del Plan de Pase a Producción.
- p. Ejecución del Plan de Pase a Producción.

Posteriormente, me desempeñé como Analista de 1° nivel de Incidencias cumpliendo las siguientes funciones:

- a. Analista encargada del diagnóstico y seguimiento de Incidencias de primer nivel.
- b. Aseguramiento del cumplimiento de tiempos de atención SLA.
- c. Aseguramiento del flujo de atención.
- d. Actualización de la matriz de escalamiento y Gestión de Conocimiento.
- e. Derivación de incidencias a diversos grupos resolutores proporcionando información recogida del monitoreo de servicios.
- f. Documentación técnica a nivel de la capa de servicios ASO y *Backend* del negocio.

2.6 Experiencia profesional ejecutada en la organización

Actualmente la profesional y autora de este trabajo de suficiencia profesional cumple las funciones de Analista de 1° nivel de incidencias y forma parte del Mantenimiento de Aplicaciones en la entidad bancaria desde noviembre del año 2020.

Previamente, la profesional cumplía las funciones de Gestora de Servicios REST y Analista del Aseguramiento de la Calidad en el Área de Arquitectura de Servicios en la misma entidad bancaria. Dicha experiencia se desarrolló desde enero del 2019 a octubre del año 2020.

Esta experiencia le ha permitido desarrollar diferentes habilidades tanto técnicas como blandas tal así que su crecimiento ha sido constante de tal modo que ha ganado la confianza de sus líderes para nuevos retos.

Capítulo III

Actividades desarrolladas

Para entender de manera completa la definición del problema y su solución, es fundamental describir los términos que se indican a continuación:

Servicio Web REST:

El estado representacional de transferencia es un tipo de arquitectura para desarrollo web de sistemas distribuidos que emplea el protocolo HTTP. REST es una de las arquitecturas más utilizadas al implementar servicios web ligeros y de fácil consumo entre clientes y el servidor.(Mejía-Granda, 2018)

La Transferencia de Estado Representacional (REST - RepresentationalState Transfer) fue ganando más reconocimiento por ser una alternativa más simple a SOAP y a los servicios web basados en el Language de Descripción de Servicios Web (Web ServicesDescripcionLanguage - WSDL) (De Seta, Leonardo, 2018)

REST hace que los desarrolladores usen los métodos HTTP de manera explícita, de tal modo que resulte consistente con la definición del protocolo. Este principio de diseño básico establece una asociación uno-a-uno entre las operaciones de crear, leer, actualizar y borrar y los métodos HTTP. De acuerdo con esta asociación: (De Seta, Leonardo, 2018)

- Se usa POST para crear un recurso en el servidor.
- Se usa GET para obtener un recurso.
- Se usa PUT para cambiar el estado de un recurso o actualizarlo.
- Se usa DELETE para eliminar un recurso.

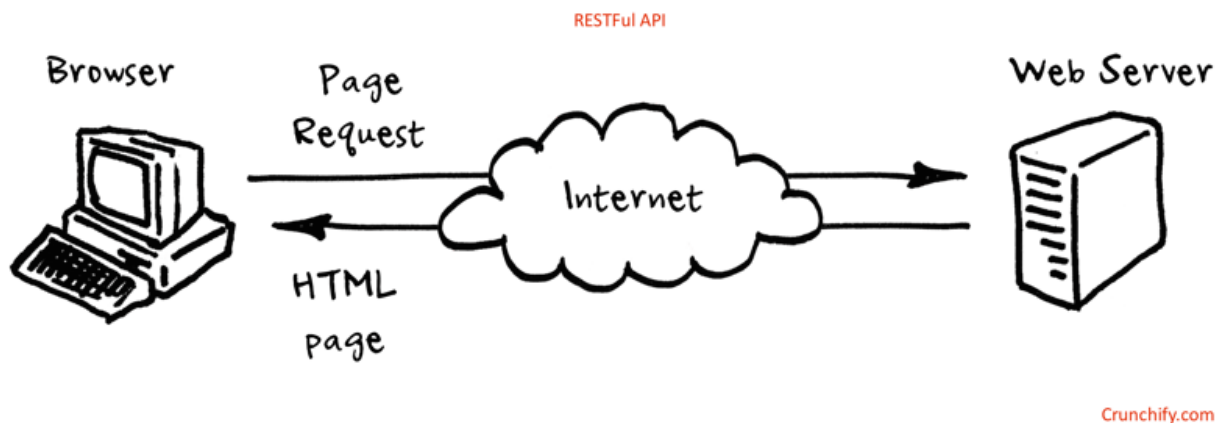


Figura 5: Comunicación Cliente - Servidor de Servicios REST.
Fuente: (solocodigoweb.com, 2016)

Ejemplo de una llamada básica entre un cliente y un Servicio REST:

Tal como es conocido la comunicación entre un cliente y un servidor es mediante peticiones y respuestas HTTP como se puede apreciar en la figura 6.

En la petición se especifica el método determinado, que puede ser GET, POST, PUT, DELETE, entre otros, de este modo se indica la acción a realizar sobre un recurso del servicio. En el ejemplo que se ve a continuación se realiza un GET sobre un usuario determinado obteniendo de esta manera la información del usuario.

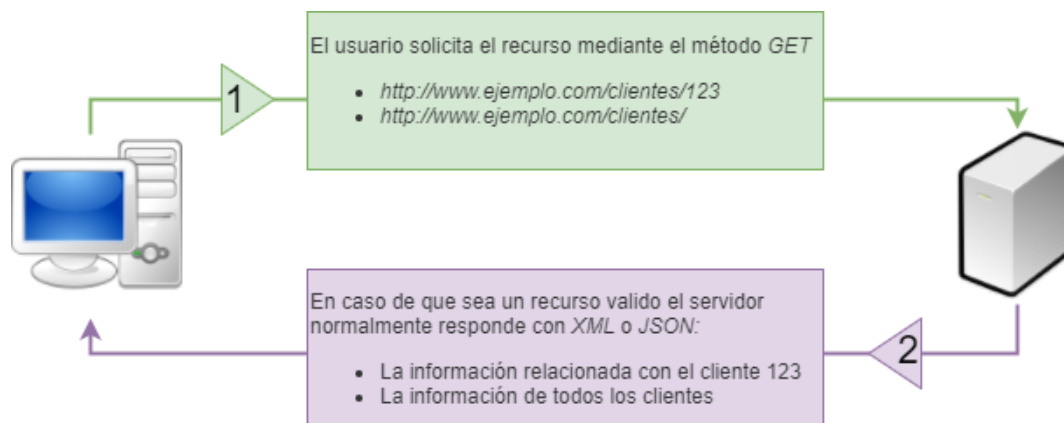


Figura 6: Operaciones con método GET.

Fuente: (blog.bi-geek.com, 2018)

Operaciones RESTful:

Existen diferentes métodos tal como son GET para obtener o leer, POST para enviar o crear y para actualizar usamos PATCH o PUT.

Tabla 6: Operaciones RESTful.

Fuente: (blog.bi-geek.com, 2018)

Operación	Método HTTP	URI	Parámetros	Resultado
Listar	GET	/ {recurso}	No aplica	Lista del tipo de recurso
Crear	POST	/ {recurso}	Dentro del cuerpo en el POST	Se crea un nuevo recurso
Leer	GET	/ {recurso}/ {recurso_id}	No aplica	Recurso en función al id
Actualizar	PATCH/PUT	/ {recurso}/ {recurso_id}	Se pasan usando una cadena de consulta	Se actualiza/reemplaza el recurso
Borrar	DELETE	/ {recurso}/ {recurso_id}	No aplica	Se elimina el recurso en función al id

Gestión de Servicios Web REST:

La gestión de servicios Web se realiza a través de un conjunto de capacidades de gestión que permiten la monitorización, el control y la generación de informes sobre la calidad y el uso del servicio. (Puerta, 2015)

Framework:

También llamado entorno de trabajo es un grupo de definiciones, buenas prácticas y criterios enfocados en atender una problemática existente. La implementación de un *framework* o marco de trabajo conlleva a la atención de problemas que tiene causas similares.

Canal:

También conocido como canal de distribución, es el medio físico o virtual por el cual un banco brinda sus servicios.

Por ejemplo, cajeros automáticos, Banca por internet, Banca Móvil, Beneficios, etc. A través de estos canales es posible transferir dinero, pagar servicios, retirar efectivo, entre otros.

CANALES BANCARIOS

Canales de Distribución

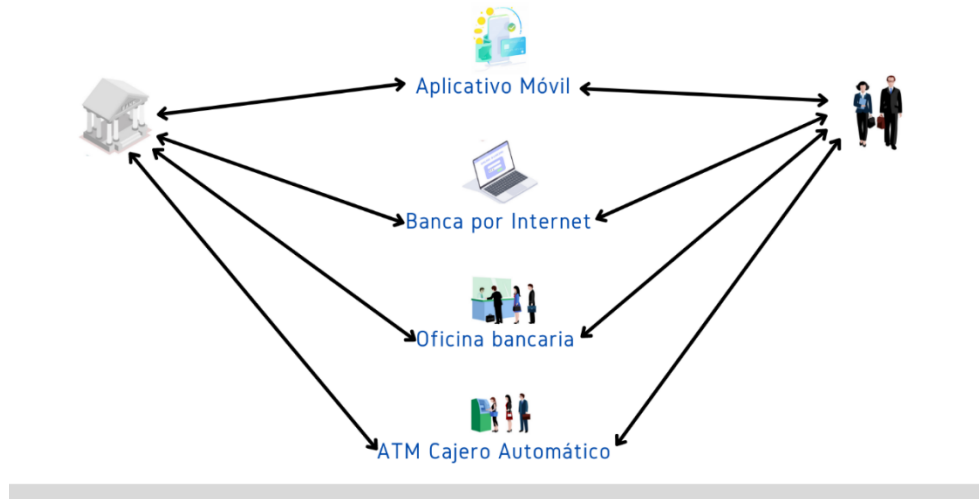


Figura 7: Canales de distribución.

Fuente: Elaboración propia

API:

Significa interfaz de programación de aplicaciones. Según RedHAT menciona que las APIs “permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados.

Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero”.

Bodyshopping:

Trabajador que, como personal externo, desempeña sus labores como parte de la entidad bancaria.

Rollback:

Llamado también reversa o marcha atrás, y se define como el procedimiento o pasos que se siguen con rigurosidad para regresar a un estado previo a un cambio, dicho estado debe ser estable y seguro.

Estos pasos son puestos en marcha cuando se realizan cambios o pases a producción fallidos. Es importante que al ejecutar el *rollback* se asegure la alta disponibilidad de los servicios.

Área de Arquitectura de Servicios:

Área de *EngineeringSolution* que tiene gobernabilidad de toda la capa de Servicios, esta capa sirve de intermediaria entre la capa *frontend* y *backend*.

Servicio ASO:

Servicio REST desarrollado como parte de la Arquitectura de Servicios.

Servicio Nuevo:

Son requerimientos donde se especifica la creación de un servicio multicanal nuevo, como parte del requerimiento se debe incluir ciertas especificaciones técnicas como el RAML, descriptores, ficha técnica del servicio y del *backend* que será consumido. Para este tipo de requerimiento no es necesario las Pruebas de No Impacto ya que al ser un servicio nuevo no es usado por otro canal que no sea el canal solicitante de su creación.

Servicio Modificado:

Son requerimientos donde se especifica la modificación de un servicio REST que ya existe en el catálogo de Servicios. Se debe indicar únicamente aquellas especificaciones que contienen el detalle de la modificación. Para este tipo de alcance es importante poner mayor interés en las Pruebas de No Impacto, ya que si la modificación solicitada por un canal impacta de manera negativa la funcionalidad de un servicio usado por otros canales se deberá desestimar el cambio.

Servicio Reuso:

O también llamado Requerimiento de Catalogación de servicios. Es el cumplimiento de una configuración externa que habilita mecanismos de seguridad con la finalidad que un canal pueda reusar un servicio, es decir, que cierto servicio pueda ser consumido desde un canal solicitante. Mayormente, para este tipo de requerimiento no requiere de Pruebas de No Impacto

ya que una configuración externa que concierne a un sólo canal no debería impactar a otros canales.

Ciclo de Vida de los Servicios REST:

Las diferentes fases en el ciclo de vida de los servicios ASO,

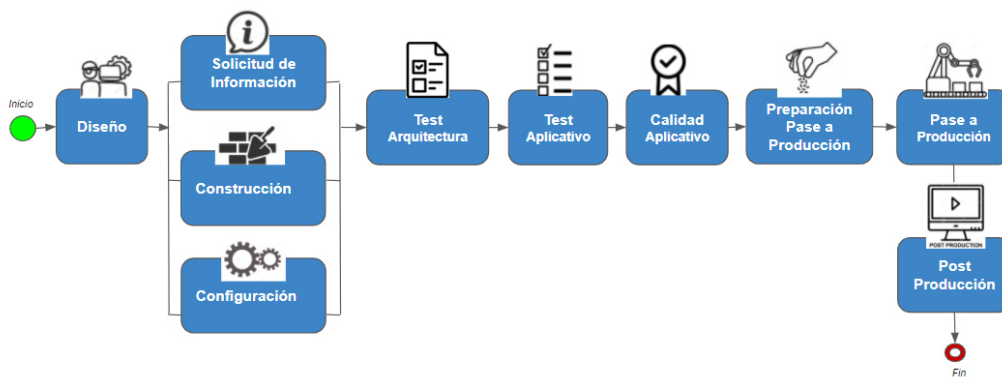


Figura 8: Ciclo de vida de Servicios REST.
Fuente: Elaboración propia.

Catalogación:

Configuración Externa que se realiza en la arquitectura con la finalidad que un canal pueda reusar un servicio multicanal existente. Además, como para de esta configuración se deben dar de alta configuraciones de seguridad.

Reuso:

Alcance de un requerimiento que consiste en la solicitud de un canal para usar un servicio productivo con la finalidad de agregarlo a su catálogo de servicios. Este alcance no incluye modificaciones ni configuraciones adicionales, sino que únicamente la catalogación del servicio para el canal solicitante.

ANS:

O en inglés SLA, tal como indica sus siglas son acuerdos de nivel de servicio, es decir, es un documento en el que se definen los acuerdos sobre las normas de un producto o servicio

que se brindará según las necesidades de un cliente. Este acuerdo encierra especificaciones y tiempos de atención.

API REST:

También llamado API de transferencia de estado representacional, es una interfaz de programación de aplicaciones la cual hace posible la interacción con los servicios web de RESTful.

También se considera como el contrato entre el proveedor de información y el usuario. En este contrato se define la respuesta que se requiere del consumidor (la petición) y el que requiere el productor (la respuesta).

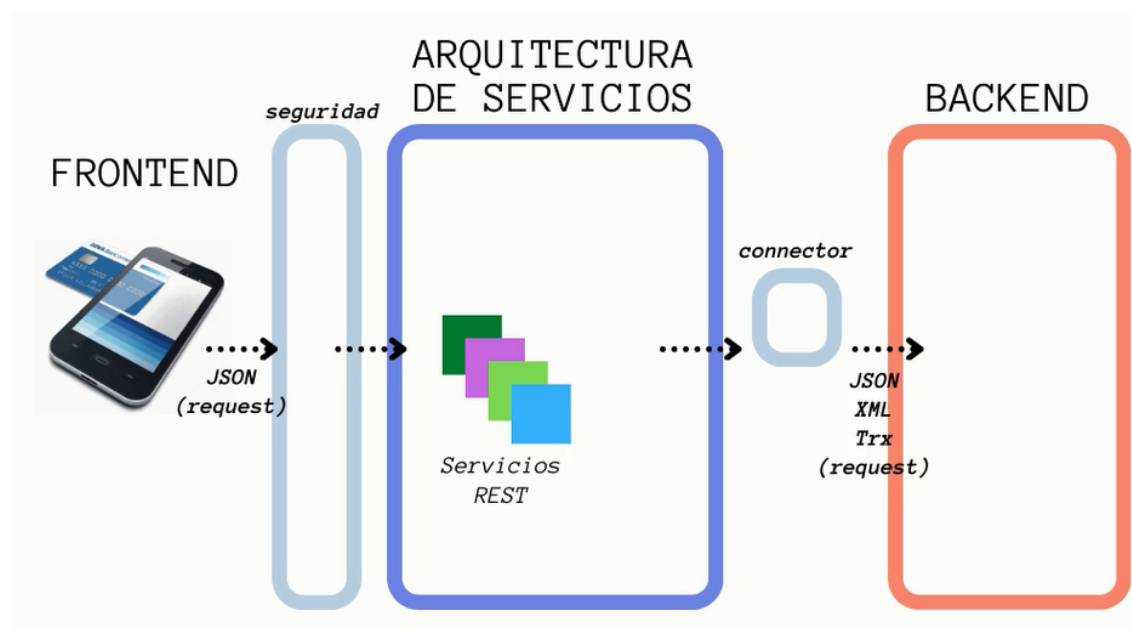


Figura 9: Arquitectura de servicios.

Fuente: Elaboración propia

ASO:

Tiene como objetivo facilitar la integración de la aplicación (generalmente canales frontales) con la lógica empresarial. Utiliza un modelo API en un entorno equipado con las herramientas necesarias de desarrollo y consumo de servicios.

Tiene las siguientes características:

- Multicanal: Permite reutilizar la lógica empresarial en diferentes canales.
- Consejos de mejores prácticas: Integra un modelo de seguridad de un extremo a otro.
- Plataforma de ejecución: Los desarrolladores se centran en la lógica empresarial que permite un desarrollo rápido.

Servicio multicanal:

Servicio REST que es consumido por diferentes canales.

Backend o back end:

Conforma la parte interna de las aplicaciones que se aloja en un servidor y se alimenta de una base de datos. En el *backend* se aloja la lógica del negocio, y provee de soluciones para que toda acción realizada en las aplicaciones se ejecute de manera correcta.

3.1 Situación Problemática

Los clientes de un banco buscan realizar con total seguridad diferentes operaciones a través de los canales de distribución con la finalidad de transferir dinero, revisar su estado de cuenta, pagar tarjetas de crédito, configurar sus cuentas, crear cuentas de ahorro a plazo fijo, abrir una cuenta en dólares, realizar cambio de divisas, entre otras más operaciones.

Dichas operaciones requieren de diferentes configuraciones para garantizar la seguridad de estas y la integridad de la información de sus clientes. Además, asegurar la alta disponibilidad a nivel tecnológico de los recursos que hacen posible el correcto funcionamiento de las operaciones. Entre esos recursos se encuentran los Servicios Web REST.

En el año 2016 el banco no contaba con un área local para la gestión de los Servicios REST así que a partir de esta necesidad es contratado los servicios de ENTELGY para la gestión de dichos servicios REST.

La sede principal del banco (España) entregó la gobernabilidad de estos servicios para que el banco en Perú pueda gestionar sus propios servicios REST. Así, inicia para ENTELGY esta labor y para ello se conforma un grupo pequeño de tres personas que optaron las funciones de *tester* de estos servicios REST y el pase a producción de estos. Hasta enero del año 2019 en el que se revisan los altos índices de pases a producción fallidos y se analizan otros problemas referentes a la gestión del equipo, dando así inicio a este proyecto que se describe en este trabajo.

3.1.1 Definición del problema

En el desarrollo, integración y puesta en producción de Servicios REST como parte del área de Arquitectura de Servicios dentro de la entidad bancaria, se venían generando un sinnúmero de requerimientos de proyectos que se atendían de manera inexacta y lenta. Dicho esto, el área recibía cuantiosas quejas de parte de gerencia *por los altos índices de pases a producción fallidas y por la atención de tareas que no correspondían al alcance del equipo en cuanto a funciones y tiempos de respuesta*. Esto traía como consecuencia un bajo nivel en el servicio y frecuentes incumplimientos en los términos de los contratos establecidos entre ENTELGY con la entidad bancaria.

En esta problemática se planteó la siguiente interrogante: ¿cuál es el motivo principal que produce una mala e ineficiente gestión de los servicios REST, lo cual produce altos índices de pases a producción fallidos?

Problemas Específicos

¿Cuál es el porcentaje de pases fallidos a producción?

¿Cada miembro del área conoce su rol, tareas y tiempos de atención?

¿Cada miembro cuenta con el mismo enfoque deservicio que brindaEntelgy?

¿Conocen los flujos que conforman los procesos relevantes al ciclo de vida de servicios Web REST?

3.2 Solución

De este modo, se llevó a cabo como solución un *framework* para mejorar de gestión de los pases a producción de servicios web REST de la entidad bancaria.

El diseño parte de un análisis exhaustivo y la implementación define la integración de “buenas prácticas” en un marco ágil integrando metodologías conocidas como son: ISTQB, Kanban, SCRUM, entre otras. Así como también la definición de procesos, funciones, roles y lo más importante tiempos de atención.



Figura 10: Problema y soluciones propuestas.
Fuente: Elaboración propia.

3.2.1 Objetivos

Implementar un *framework* para mejorar la gestión de pases a producción de Servicios Web REST que sirva como marco para la atención de requerimientos a lo largo del ciclo de vida de estos servicios Web REST, tal como es la atención de configuraciones y detección temprana de fallas en los diferentes Servicios Multicanales tanto a nivel de desarrollo (código) como también en las configuraciones externas (integración). Finalmente, la puesta en producción, de

manera que, haya una mejor calidad en la atención en cuanto a tiempo, cantidad de bugs detectados y lo más importante en cuanto a la disminución de pases a producción fallidos.

Objetivos específicos

Los objetivos específicos se centran en la problemática que en este trabajo se presenta, tal y como son:

- Realizar un análisis detallado sobre la cantidad de *rollbacks*, tiempo de atención y motivos de *rollback*.
- Definir los roles, funciones y los tiempos de atención para cada una de esas tareas específicas.
- Llevar a cabo un plan de reuniones y buenas prácticas que permitan el control y seguimiento de las diferentes tareas.
- Mapear los procesos y su interacción con las diferentes dependencias y sus tiempos de respuesta.

3.2.2 Alcance

Alcance Funcional:

El alcance funcional del proyecto incluye las siguientes fases en el Ciclo de Vida de los servicios multicanales, los cuales serán soportados por el *framework* propuesto. Dicho de otra manera, el *framework* tendrá un impacto en las fases señaladas en la figura 11.

Este *framework* no incluye las tareas de desarrollo o construcción de Servicios REST ni pruebas realizadas por el Equipo SCRUM llamado también Calidad Aplicativo (ver imagen)

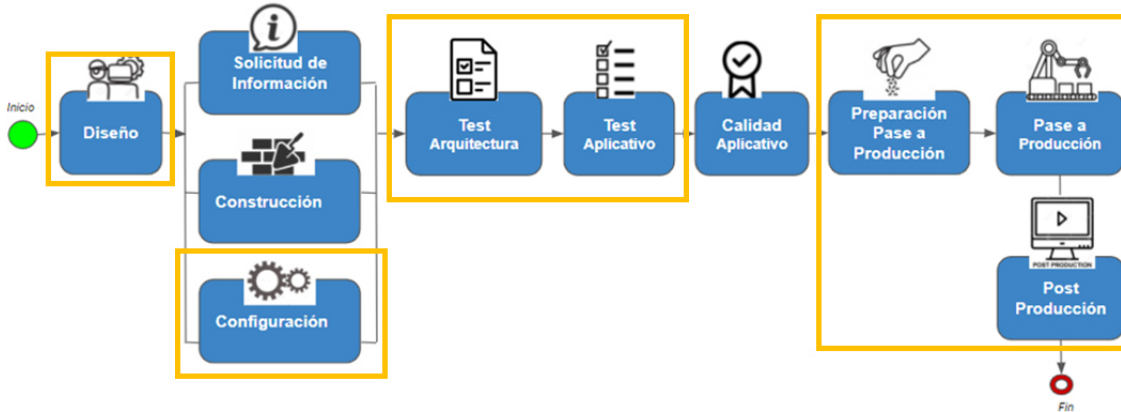


Figura 11: Flujo General del Ciclo de Vida de Servicios REST.
Fuente: Elaboración propia.

Alcance Organizacional

El alcance organizacional del proyecto abarca la subárea de la Arquitectura de Servicios ASO. Dicha subárea se encuentra dentro del Área de Ingeniería de la entidad bancaria.

Alcance Geográfico

El diseño y gestión de Servicios REST que se desarrollan de manera local únicamente tendrá un impacto a nivel nacional.

3.2.3 Etapas y Metodologías

ENELGY es una empresa que se adapta a las distintas metodologías y *frameworks* de sus clientes, sin embargo, como parte del servicio que ENELGY brinda es la de proponer diferentes metodologías y buenas prácticas con la que ENELGY asegura la calidad del servicio que recibe el cliente. Para fines de este trabajo se detalla como marco de trabajo diferentes prácticas, flujos y procesos que integran diferentes metodologías enfocadas en procesos e incidencias. Dichas metodologías se detallan en la página 50.

El *core* de este *framework* es, principalmente, las funciones, los perfiles, los procesos y finalmente los indicadores. Cada uno de estos componentes ataca directamente las falencias anteriormente descritas.

3.2.3.1 Etapas:

Las etapas que abarca el proyecto son las siguientes:

1. Estructura del Framework:



Figura 12: Diagrama Fases del Framework.
Fuente: Elaboración propia.

Mediciones

En esta fase se debe tener en cuenta el análisis situacional, para ello se recogió información concerniente a los procesos, roles, funciones y tiempos de atención para cada una de las fases del Ciclo de vida de los Servicios REST (o también llamados Servicios Multicanales)

Es importante partir por levantar información relevante sobre el estado situacional del área. Conocer qué es lo que llevó al equipo a la situación en la que ahora se encuentra y que ha dado como resultado la problemática.

Esta fase tiene como objetivo detectar los puntos de dolor y sus posibles causas. Además, se debe indicar los potenciales problemas que nos encontraríamos si estos problemas no se atienden a tiempo.

Funciones y Roles

Esta fase tiene como objetivo definir las funciones e identificar el o los roles que intervienen en las diferentes fases en el ciclo de vida de los Servicios REST.

Ejecuciones de tareas por tiempo de atención

En esta fase se determinan los acuerdos de nivel de servicio en cuanto a tiempo de atención que se consideraron para la ejecución de cada tarea. Se debe tener en cuenta que cada fase en el ciclo de vida está definidas las tareas (véase en la sección 3.2.5)

Entregables

Es importante definir y diseñar los entregables que aportarán valor al proyecto. Un entregable puede ser definido como el resultante de un proceso, puede ser un producto, un documento o un servicio. El objetivo de esta fase es definir qué entregables serán proporcionados, quién o quiénes están asignados para elaborar el entregable y en qué parte del proceso serán generados.

Refinamiento

Para esta última fase es necesario indicar la importancia de las reuniones de retrospectiva y revisión ya que a partir de esa actividad se busca realizar mejoras en todos los aspectos del *framework*, es decir, procesos, diagramas, documentación y herramientas.

2. Implementación del Framework:

Es importante mencionar que la implementación de este *framework* se ejecutó de manera gradual, de ahí la importancia de una adecuación continua.

Mediciones – Implementación:

Gracias a un análisis profundo acerca de los procesos se identificó los siguientes puntos que fueron atendidos en la solución:

a) Mala gestión de recursos:

El trabajo realizado por el equipo de gestores de servicios está compuesto por tareas que se realizan a demanda, tal es así que los proyectos iniciaban en cualquier momento del año sin planificación previa. Esto traía consigo un problema importante de tiempos, presupuestos y recursos.

Para ilustrar mejor este punto, se encontró que el equipo de Gestores tenía meses con sobrecarga de trabajo y en otros meses sin ninguna tarea asignada. Esta situación revela la falta de gestión en el área.

El responsable del área se vio obligado a solicitar más recursos para poder cumplir con toda la demanda de trabajo comprometido por los equipos SCRUM. Esta acción trajo consigo otros problemas como por ejemplo el no considerar la curva de aprendizaje de los nuevos integrantes, ya que aun contratando a personal calificado éstos tenían que pasar por capacitaciones extensas para entender la arquitectura, los procesos y los roles del área. En poco tiempo, cuando la demanda disminuía, el recurso personal se quedaba sin tareas asignadas. Lo que se traduce como una pérdida en el presupuesto.

Para poder atender esta problemática se acordó que todos los proyectos SCRUM deben desarrollarse dentro de PIs o también llamado ProgramIncrement que se define como una ventana de tiempo en la que se realiza un desarrollo o implementación el cual proporciona un incremento de valor. Además, se definen 4 PIs por año, es decir, todos los proyectos son programados en cada PI que tendrá una duración de 3 meses.

Los proyectos SCRUM tienen en este periodo el alcance que crean conveniente según su proyección.

b) Funciones no definidas:

El problema de las funciones no especificadas y los roles no identificados ocasionó que se derivan tareas al área sin que ellos tenga las herramientas, el conocimiento y los permisos para efectuarla. Sin embargo, cuando una tarea de tal naturaleza era derivada al área los gestores no tenían la autoridad para rechazarla. Esto creó muchos inconvenientes entre colegas y lo que es peor tareas olvidadas o no atendidas. Lo que trae un impacto importante ya que ocasionó retrasos en los Equipos SCRUM.

Sobre este punto se estableció se definieron los roles, las funciones que tendría cada rol, especificación de sus tareas y los tiempos de atención según la naturaleza del requerimiento.

Funciones y Roles – Implementación:

Se define que los recursos cumplan con las siguientes funciones que son ejecutadas a lo largo del Ciclo de Vida de los Servicios REST. Se ha descrito estas funciones teniendo en cuenta los roles de **Diseñador** y **Gestor de Servicios**.

El Diseñador de Servicios REST es responsable de:

Proponer y diseñar soluciones en concepto de Servicios REST para proyectos SCRUM.
Tener la gobernabilidad de los documentos técnicos de los servicios multicanal.

Funciones del Diseñador:

1. Trabajar en conjunto con el equipo SCRUM para diseñar la solución que cumpla los requerimientos del proyecto.
2. Propone mejoras teniendo en cuenta sus conocimientos de los diferentes servicios y canales.
3. Proporciona además un documento llamado Ficha Técnica del Servicio Multicanal o Documento SMC.
4. Finalmente, proporciona apoyo inicial al Gestor del Servicio.

El Gestor del Servicio ASO es responsable de:

Gestionar servicios multicanales y dar soporte técnico a los equipos SCRUM en cuanto al funcionamiento de los servicios multicanal.

Funciones del Gestor de Servicios:

1. A inicios de cada PI, envía un correo al Equipo SCRUM solicitando información sobre el proyecto y los servicios que se atenderán como parte del mismo. Esto es importante para que el Gestor tenga mapeado todos los servicios y sus respectivos alcances (servicio nuevo, servicio modificado, *reuso* de servicio)
2. Realiza el despliegue de los componentes en los diferentes entornos a través de *Jenkins*.
3. Configura los parámetros del servicio, así como la comunicación con el *backend*.
4. Prueba los servicios solicitados con los datos de prueba facilitados por el Equipo SCRUM.
5. Identifica y reporta problemas o incidencias sobre sus pruebas.
6. Realizar el T907, documento o entregable en el que el Gestor evidencia el correcto funcionamiento del servicio.
7. Atiende, analiza y determina las incidencias presentadas durante este tiempo.
8. Acompaña y resuelve las dudas del Equipo SCRUM durante sus pruebas.
9. Coordina y realiza el despliegue de los componentes al entorno de Calidad bajo aprobación del líder de gestores. Es importante contar con la aprobación para asegurar la estabilidad del ambiente.

10. Atiende, analiza y determina las incidencias presentadas durante este tiempo en el ambiente de Calidad.
11. Acompaña y resuelve las dudas del Equipo SCRUM durante sus pruebas en el ambiente de Calidad.
12. Envía la lista de Canales impactados al Equipo SCRUM para que coordine la ventana de pruebas. Esta tarea es de suma importancia para todos aquellos equipos SCRUM que tenga programado un pase a Producción. Ha de entenderse que si las pruebas de No Impacto no fueran exitosas no se podría agendar el pase a Producción.
13. Solicita al equipo de Arquitectura de Servicios Operaciones los componentes integrados para la ventana de pruebas. Estos componentes deberán ser el mismo que pase a producción.
14. De ser el caso, y con el apoyo del Equipo SCRUM, prepara una ventana de pruebas preliminar en el entorno de Test con los componentes integrados para asegurar su correcta configuración.
15. Realiza el seguimiento de la ventana de pruebas de no impacto para responder lo antes posible cualquier observación o incidencia de los Canales impactados.
16. Crea las hojas de pase a producción llamadas Hojas P110 el cual deberá ser firmado (con firma digital) por todos los interesados, es decir, el responsable del área de Arquitectura de Servicios ASO, responsable de Gestión de Cambios y Scrum Master. Además, el gestor deberá crear un ticket en JIRA para el pase llamado también Ticket *release* JIRA.
17. Realiza la gestión para la configuración del entorno Productivo (*backend*, cifrado, catalogación, alta de URL *backend*, alta de propiedades, entre otros).
18. Prepara los componentes para su despliegue al entorno de Producción, esta tarea se realiza a través de *Jenkins*.
19. Envía un correo a los involucrados en el pase para las pruebas de no impacto en los entornos de Liga Oculta (pre-producción) y Producción.
20. En caso no se cuente con el visto bueno de todos los canales impactados al finalizar el pase, revierte los cambios desplegados en el entorno de Producción. Esta tarea es llamada *rollback*.
21. Realiza el seguimiento de las pruebas de no impacto.

22. Analiza y atiende de ser necesario las incidencias reportadas en el post pase a producción.

Esta tarea es llevada a cabo en las siguientes 48 horas después de un pase a producción.

Si pasado ese tiempo se evidencia algún comportamiento no deseado en el servicio esto será atendido por el área de operaciones (subárea de apoyo transversal)

Ejecuciones de tareas por tiempo de atención – Implementación:

En cada una de las fases del ciclo de vida de los servicios REST se cumplen ciertas tareas (que están definidas en el capítulo 3.2.5) Dichas tareas son definidas teniendo en cuenta el tiempo que requiere para su ejecución. A continuación, se detalla el tiempo de atención por cada fase y cada tipo de alcance de requerimiento.

Tabla 7 : Fases por Tiempo de Atención.
Fuente: Elaboración propia.

Fase en el Ciclo de Vida de un Servicio ASO	Alcance del Requerimiento			
		<i>NUEVO</i>	<i>MODIFICACIÓN</i>	<i>REUSO</i>
	Diseño	5 días laborales	3 días laborales	4 horas
	Construcción	3 días laborales	1 día laboral	NA
	Configuración	4 horas	4 horas configuración	3 horas
		configuración local	local	configuración
		3 días laborales	3 días laborales	local
		configuraciones	configuraciones	
		holding (México)	holding (México)	
	Test Arquitectura	3 días laborales	1 día laboral	2 horas
	Test Aplicativo	NA*	NA*	NA*
	Calidad	Tiempo que sea requerido por el Equipo Scrum. No debe superar el periodo de un PI (3 meses)		
	Preparación	5 días laborales, incluye:		
	Pase a	Pruebas de No Impacto (1 día)		
	Producción	Configuraciones Externas (3 días)		
		Aprobación de Cambio (1 día)		
	Pase a	Pre-Producción: 3 horas (06:00 p.m. a 09:00 p.m. Hora México)		
	Producción	Producción: 4 horas (11:00 p.m. a 03:00 a.m. Hora México)		
		NOTA: Es posible tener una extensión de 2 horas adicionales		
	Post	48 horas posteriores al cambio. Incluye:		
	Producción	Monitoreo de servicios.		
		Revisión proactiva de disponibilidad y estabilidad del servicio.		
		Soporte técnico y funcionales al Equipo Scrum.		

* El ANS no contempla la fase de Test Aplicativo ya que se trata de tareas relacionadas al Equipo SCRUM y es este equipo el que tiene la libertad de sus propios Planes de Prueba y Despliegue.

Entregables – Implementación:

Los entregables que se muestran a continuación ya existían como parte de la documentación que generaba el área, para este proyecto se realizaron mejoras en cuanto a la presentación y contenido de estos.

Siendo así, los más principales el T907 y P110.

T907: Más conocido como el documento donde se especifican los casos de prueba y sus resultados, en pocas palabras donde se detallan las evidencias del correcto funcionamiento de un servicio REST. Este documento es entregado al finalizar las pruebas funcionales en la fase de Test Arquitectura. Y, además, es el documento que ayudará de guía y garantía para que el Equipo Scrum ejecute el Plan de Pruebas, teniendo en cuenta que, al no obtener resultados esperados podrá deducir que no se trata de un defecto en la configuración, o a nivel de desarrollo del servicio.

AAP: 13000043

Create Ticket:

Cliente REST: Postman

URL: http://150.2[REDACTED]

Evidencia solicitud: Petición realizada en la herramienta Postman

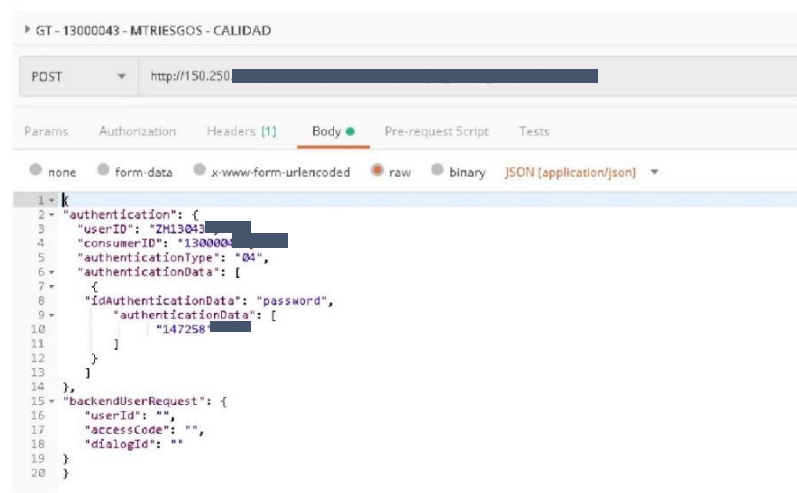


Figura 13: Modelo de T907.

Fuente: Elaboración propia

P110: También conocido como Hoja de Pase a Producción, es un documento en el que se especifica el componente que se desea desplegar en ambientes productivos. En este documento se incluyen datos del Proyecto, servicios, alcances de cada uno de los servicios y firmado por el responsable del área, el Scrum Master del proyecto, y el encargado de Gestión de Cambios.

FECHA dd/mm/aaaa																																											
PASE A PRODUCCIÓN																																											
HOST	DISTRIB X																																										
CÓDIGO IDENTIFICADOR																																											
APLICATIVO / EQUIPO	PaaS																																										
LIDER RELIABILITY/DISCIPLINA/ESPECIALIDAD SYSTEMS	Arquitectura de Servicios-JBOSS																																										
DESCRIPCIÓN	Pase GESTCOTIZA - PRE PRODUCCIÓN - Servicios nuevos y reusos, se requiere catalogación y reinicio de instancias.																																										
<table border="1" style="width: 100%; border-collapse: collapse; font-size: 0.8em;"> <thead> <tr style="background-color: #e6f2ff;"> <th>LENGUAJE / PAQUETE</th> <th>COMPONENTE</th> <th>TIPO DE COMPONENTE</th> <th>REFERENCIA / MOTIVO (Agregar líneas si lo requiere)</th> <th>TIPO</th> <th>CAMBIO</th> <th>REQUERIMIENTO</th> </tr> </thead> <tbody> <tr> <td>PAQUETE</td> <td>non_businesses</td> <td>JOB JENKINS</td> <td>https://</td> <td></td> <td>1755748</td> <td></td> </tr> <tr> <td>PAQUETE</td> <td>non_customers</td> <td>JOB JENKINS</td> <td>https://</td> <td></td> <td>1755748</td> <td></td> </tr> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> <tr> <td colspan="3" style="background-color: #e6f2ff; font-size: 0.7em;">ESPECIALISTA SYSTEMS</td> <td style="font-size: 0.7em;">REGISTRO:</td> <td>P020187</td> <td style="background-color: #e6f2ff; font-size: 0.7em;">ANEXO:</td> <td>1039</td> </tr> <tr> <td colspan="2" style="background-color: #e6f2ff; font-size: 0.7em;">OBSERVACIONES:</td> <td colspan="5" style="font-size: 0.7em;"> /createnonbusinessescontacts (REUSO) /getnonbusinessescontact (NUEVO) /modifynonbusinessescontact (NUEVO) /deletenonbusinessescontact (NUEVO) /createnonbusiness (REUSO) /listnonbusinesses (NUEVO) /deletenonbusiness (NUEVO) /modifynonbusiness (NUEVO) /listbusiness (REUSO) /createnoncustomer (NUEVO) /getnoncustomer (NUEVO) /deletenoncustomer (NUEVO) /listnoncustomers (NUEVO) </td> </tr> </tbody> </table>		LENGUAJE / PAQUETE	COMPONENTE	TIPO DE COMPONENTE	REFERENCIA / MOTIVO (Agregar líneas si lo requiere)	TIPO	CAMBIO	REQUERIMIENTO	PAQUETE	non_businesses	JOB JENKINS	https://		1755748		PAQUETE	non_customers	JOB JENKINS	https://		1755748									ESPECIALISTA SYSTEMS			REGISTRO:	P020187	ANEXO:	1039	OBSERVACIONES:		/createnonbusinessescontacts (REUSO) /getnonbusinessescontact (NUEVO) /modifynonbusinessescontact (NUEVO) /deletenonbusinessescontact (NUEVO) /createnonbusiness (REUSO) /listnonbusinesses (NUEVO) /deletenonbusiness (NUEVO) /modifynonbusiness (NUEVO) /listbusiness (REUSO) /createnoncustomer (NUEVO) /getnoncustomer (NUEVO) /deletenoncustomer (NUEVO) /listnoncustomers (NUEVO)				
LENGUAJE / PAQUETE	COMPONENTE	TIPO DE COMPONENTE	REFERENCIA / MOTIVO (Agregar líneas si lo requiere)	TIPO	CAMBIO	REQUERIMIENTO																																					
PAQUETE	non_businesses	JOB JENKINS	https://		1755748																																						
PAQUETE	non_customers	JOB JENKINS	https://		1755748																																						
ESPECIALISTA SYSTEMS			REGISTRO:	P020187	ANEXO:	1039																																					
OBSERVACIONES:		/createnonbusinessescontacts (REUSO) /getnonbusinessescontact (NUEVO) /modifynonbusinessescontact (NUEVO) /deletenonbusinessescontact (NUEVO) /createnonbusiness (REUSO) /listnonbusinesses (NUEVO) /deletenonbusiness (NUEVO) /modifynonbusiness (NUEVO) /listbusiness (REUSO) /createnoncustomer (NUEVO) /getnoncustomer (NUEVO) /deletenoncustomer (NUEVO) /listnoncustomers (NUEVO)																																									
(Firmar aquí en caso de contingencia) <table border="1" style="width: 100%; border-collapse: collapse; font-size: 0.7em;"> <tr> <td style="width: 30%; background-color: #e6f2ff;">Nombre</td> <td></td> </tr> <tr> <td style="background-color: #e6f2ff;">Rol</td> <td>Lider: Disciplina / Reliability /Especialidad, o Scrum Master</td> </tr> </table> (Firmar aquí en caso de contingencia) <table border="1" style="width: 100%; border-collapse: collapse; font-size: 0.7em;"> <tr> <td style="width: 30%; background-color: #e6f2ff;">Nombre</td> <td></td> </tr> <tr> <td style="background-color: #e6f2ff;">Rol</td> <td>Lider QA, o Quality Coach</td> </tr> </table> (Firmar aquí en caso de contingencia) <table border="1" style="width: 100%; border-collapse: collapse; font-size: 0.7em;"> <tr> <td style="width: 30%; background-color: #e6f2ff;">Nombre</td> <td></td> </tr> <tr> <td style="background-color: #e6f2ff;">Rol</td> <td>Lider Change Mgmt.</td> </tr> </table>		Nombre		Rol	Lider: Disciplina / Reliability /Especialidad, o Scrum Master	Nombre		Rol	Lider QA, o Quality Coach	Nombre		Rol	Lider Change Mgmt.																														
Nombre																																											
Rol	Lider: Disciplina / Reliability /Especialidad, o Scrum Master																																										
Nombre																																											
Rol	Lider QA, o Quality Coach																																										
Nombre																																											
Rol	Lider Change Mgmt.																																										

Figura 14: Modelo de P110 - Hoja de Pase
Fuente: Elaboración propia.

3.2.3.2 Metodologías:

En el presente trabajo se ha tomado en consideración las siguientes metodologías:

- **Agile.** Aportará a este proyecto la flexibilidad y velocidad que requiere.
- **SCRUM.** Aunque la Gestión de Servicios REST no está enfocado a SCRUM, cada gestor brinda sus servicios a Proyectos y éstos están conformados por Equipos SCRUM.(SCRUM GUIDE, 2020)
- **Kanban.** El enfoque de la ejecución de tareas brinda a este *framework* una mejora significativa en la Gestión de Servicios ya que cada Gestor deberá cumplir cerca de 25 tareas a lo largo del ciclo de vida de un servicio REST.(KANBANIZE, 2020)
- **ISTQB Agile Testing.** Importante metodología dirigido en la detección temprana de fallas enfocado en metodologías ágiles.(IT Process Maps, 2021)

3.2.4 Fundamentos Utilizados

Para la implementación de este *framework* se incluyeron buenas prácticas en torno a la calidad de software, gestión de flujos y gestión de incidencias.

3.2.4.1 ISTQB Agile Testing

Se conoce que mientras más temprana es la detección de una falla, menos coste tendrá atenderla. De acuerdo con el tercer principio del enfoque de *ISTQB Agile Testing* entendemos que el eliminar defectos que tienen una permanencia importante en el sistema causa que el costo se incremente. Visto desde el desarrollo, continuar con la escritura de código sobre un defecto no detectado produciría que no sólo se desee levantar el defecto, sino que todo aquello que se escribió sobre él.(IT Process Maps, 2021)

Es importante que se tenga en cuenta un diccionario de defectos conocidos el cual ayudaría a la agrupación de estos, ya que, como bien se menciona en el cuarto principio de ISTQB Agile Testing, donde el *tester* encuentre un defecto es más probable encontrar más. (Mera Paz, 2017)

Además, ISTQB Agile Testing nos proporciona diferentes herramientas utilizadas para dar soporte a las distintas actividades de prueba (Mera Paz, 2017). En este *framework* se ha adoptado herramientas para la gestión de configuración haciendo uso de herramientas de *versionamiento* para APIs donde se especifica el país, la categoría, la versión, el método, la API, campos de entrada y salida sea en el *queryparam* o *pathparam*, obligatoriedad de campos, etc.

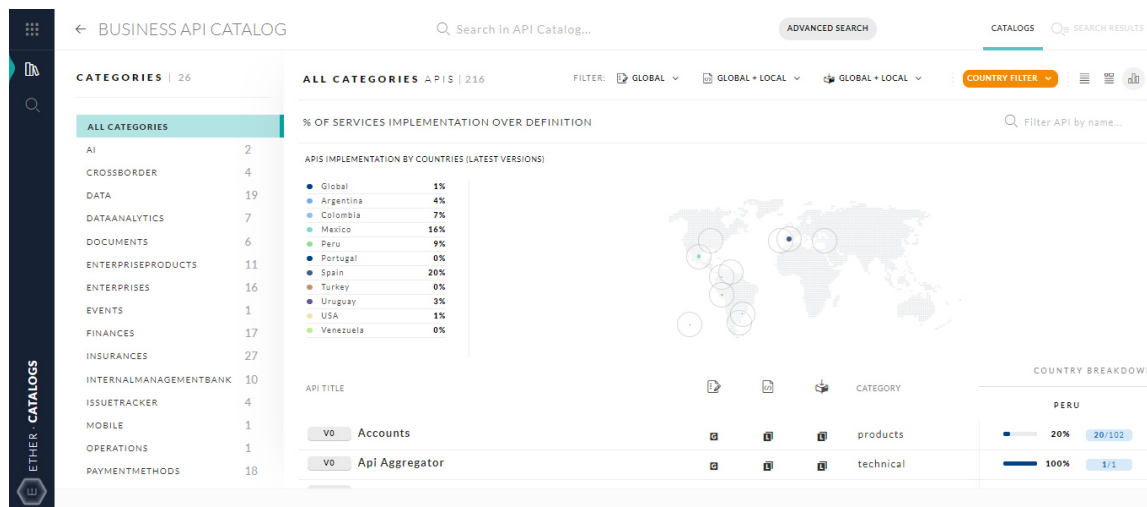


Figura 15: EtherCatalogs.

Fuente: Herramienta interna del banco (no es posible mostrar la fuente)

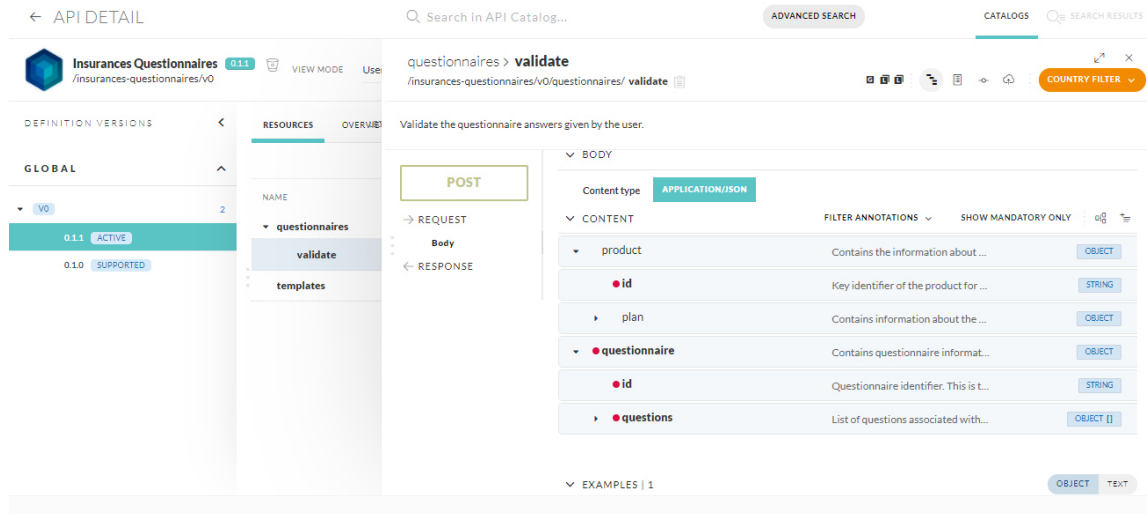


Figura 16: EtherCatalogs(otra vista)

Fuente: Herramienta interna del banco (no es posible mostrar la fuente)

En este *framework* se incluyeron algunas herramientas que el área tenía a su disposición. Estas herramientas que sirven de soporte para la ejecución del *framework*.

Tabla 8: Herramientas de soporte al framework.

Fuente: Elaboración propia.

Herramienta	Función
GITHUB	Control de Versiones.
REMEDY BMC	Gestión de Cambios.
ATENEA	Monitoreo de Servicios
JENKINS	Integración y despliegue continuo
GOOGLE APPS	Documentador, sitio, hoja de cálculo, etc.

También, se ha incluido las pautas descritas en *ISTQB Agile* sobre la ejecución de pruebas funcionales las cuales se llevan a cabo inmediatamente después que fábrica entrega los componentes que se desplegarán en el ambiente *TEST*, este ambiente está reservado únicamente para la realización de pruebas funcionales realizadas por los Gestores, llamado también **Test Arquitectura** (ver en 3.2.5 Implementación de la solución)

Se tiene definido que el Plan de Pruebas en esta fase tienen únicamente 3 casos de prueba, que son:

- **Caso de Prueba 1:** Caso de Éxito, se consideran datos de pruebas que traigan resultado esperado en una petición que lleve a cabo el objetivo por el cual fue desarrollado.
- **Caso de Prueba 2:** Caso de Error, se consideran datos de prueba que provoquen, adrede, un error no controlado en la respuesta de la petición.
- **Caso de Prueba 3:** Caso de Aviso, se consideran datos de prueba que generen un aviso en forma de error controlado y ya mapeado dentro de la Gestión de Errores del servicio.

	Descripción CP 01	Descripción CP 02	Descripción CP 03
	Describe un caso de Éxito	Describe un caso que genere Error	Describe un caso que genere un Aviso
CAMPOS DE ENTRADA	Caso 01	Caso 02	Caso 03
{model.brand.id}	403	309##	309
{model.id}	403001	309477	309477
{model.year}	2016	2014	2014

Figura 17: Ejemplo de especificación de Casos de Prueba.

Fuente: Elaboración propia.

3.2.4.2 SCRUM

Si bien es cierto, el área de Arquitectura de Servicios no opera bajo el enfoque ágil, sin embargo, se han adoptado algunas buenas prácticas ágiles de Scrum en este *framework* tal como son:

Reuniones diarias:

En el inicio de la implementación de este *framework* se celebraron reuniones de a pie, estas reuniones tenían una demora de 2 a 3 minutos por Gestor de Servicios. Debido al éxito de esta buena práctica, fue implementada como parte del *framework*. Se respondían las siguientes preguntas:

¿Qué avance se realizó? O ¿cuántos de los pendientes del día anterior se ejecutaron?

¿Cuáles son los pendientes del día de hoy? ¿cuánto tiempo me tomará llevarlo a cabo?

¿Hay bloqueantes? ¿qué tipo de bloqueante es? ¿se requiere apoyo adicional?

Reunión de Retrospectiva:

Este tipo de reuniones tenían lugar una vez a la semana o al día siguiente de algún pase crítico. El objetivo de esta reunión es que el gestor exponga con sus propias palabras y haciendo uso de su criterio lo que funcionó muy bien y lo que debería de mejorar. Estas reuniones se daban con la participación únicamente de gestores, esto era importante ya que recibíamos información más certera ya que nos encontrábamos en una reunión sin los líderes o responsables.

FECHA	ENTORNO	CRQ	GESTOR	CANALES	STATUS	BLOQUEANTES	IMPACTO	CÓMO SE RESOLVIÓ	EQUIPO RESOLUTOR	PUNTOS DE MEJORA
5/11/2019	LO	CRQ000001580039	Cecilia Riveros	GLOM OPRI	Finalizado	Hora de inicio de pase LO se retrasó por diferencia horaria con México.	Retraso es las Pruebas de LO	Se comunicó a los aplicativos que el pase LO iniciaría a las 7pm (hora Perú) y el de PROD iniciaría a las 12am (hora Perú)	SIN COMENTARIOS	México debe informar para evitar crear CRQs con hora errada.
5/11/2019	PROD	CRQ000001580194	Cecilia Riveros	GLOM OPRI	Finalizado	El soporte del canal Intraday no respondía para pruebas.	SIN COMENTARIOS	Oscar asumió la responsabilidad y envió correo autorizando el pase e indicando que Intraday probaría después.	SIN COMENTARIOS	SIN COMENTARIOS
6/11/2019	LO	CRQ000001588555	Diego de la Cruz	MNETPRI	Finalizado	Demora a la hora del reinicio.	SIN COMENTARIOS			
6/11/2019	PROD	CRQ000001588563	Diego de la Cruz	MNETPRI	Con Error	No se consideró el host name de producción cero para el Alta de proxy.	Que los servicios implicados no respondan.	Comunicandose con seguridad web para habilitar la salidad del hostname que hacia referencia a produccion cero		Considerar Producción Cero para el alta de proxys. Solicitar la verificar la comunicación. Si es con evidencia mucho mejor.
7/11/2019	LO	CRQ000001589847	Cecilia Riveros	GLOM OPRI	Cancelado		SIN COMENTARIOS	SIN COMENTARIOS	SIN COMENTARIOS	SIN COMENTARIOS
7/11/2019	PROD	CRQ000001589760	Cecilia Riveros	GLOM OPRI	Cancelado		SIN COMENTARIOS	SIN COMENTARIOS	SIN COMENTARIOS	SIN COMENTARIOS
11/11/2019	LO	CRQ000001591405	Cluny Rojas	NETCASHWEB	Finalizado	Demora en la hora de reinicio.		Comunicandose con FUP		
11/11/2019	PROD	CRQ000001591407	Cluny Rojas	NETCASHWEB	Finalizado	Error por el lado del backend Visanet (se valido al día siguiente). El aplicativo se quedo dormido. No se tenía su numero y el que tenía FUP no estaba disponible.		La validacion se realizo al día siguiente. Se llamo al aplicativo por hangout.		Solicitar numeros de contacto de los aplicativos involucrados en el pase

Figura 18: Seguimiento Retrospectiva de Post Pase a Producción.

Fuente: Elaboración propia.

3.2.4.3 KANBAN

Para la gestión de requerimientos se implementó JIRA, que es una herramienta que nos proporciona un tablero Kanban. En la figura 19 se muestra la pizarra que usó el equipo de Arquitectura de Servicios REST, como se puede apreciar cada fase de esta pizarra corresponde a una fase del ciclo de vida de los servicios y, además, tener en cuenta que cada ticket corresponde a un servicio REST o requerimiento. También se visualiza las personas asignadas, entre ellos, desarrolladores de fábrica, diseñadores y gestores de Servicios REST. Finalmente, se aprecian la criticidad de cada ticket que puede ser baja, media o alta. Con esta pizarra el equipo logra ver el estado de cada requerimiento.

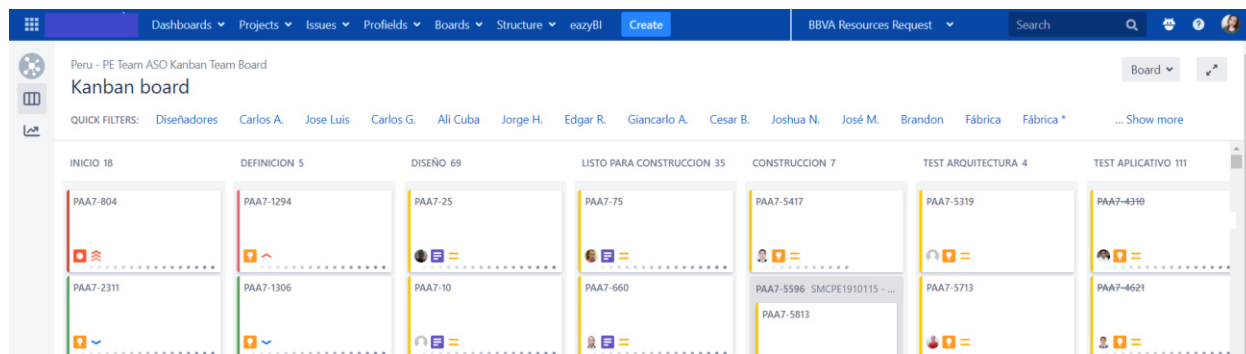


Figura 19: Jira de Arquitectura de Servicios REST.

Fuente: Peru - PE Team ASO Kanban TeamBoard(no es posible mostrar la fuente)

Se puede apreciar en la siguiente figura el contenido básico de un ticket: nombre del Servicio REST o Servicio multicanal, la prioridad, proyecto al que pertenece, la persona asignada o responsable del ticket, documentación técnica del diseño servicio, ficha técnica del servicio, y estado del ticket:

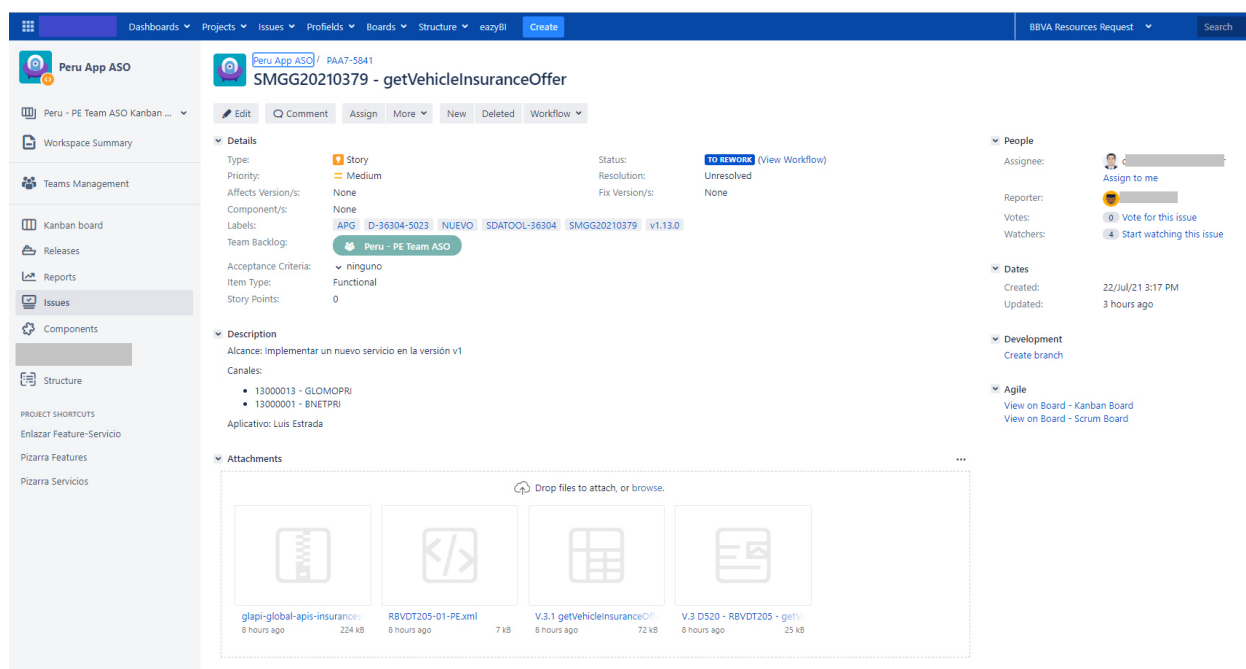


Figura 20: Especificaciones de un ticket.

Fuente: Peru - PE Team ASO Kanban TeamBoard(no es posible mostrar la fuente)

3.2.4.4 ITIL v.4

Debido a que el tipo de servicio que brinda ENTELGY se da por la relación *outsourcing*-cliente es necesario indicar que éste es dado bajo acuerdos en el nivel de servicio (ANS o SLA por sus siglas en inglés).

Estas especificaciones en el servicio son definidas por acuerdo mutuo entre empresa y cliente. Los acuerdos están dirigidos en concepto de tipo de servicio y tiempo de atención como bien lo indica ITIL para la Gestión del Nivel de Servicio.(Mera Paz, 2017)

3.2.5 Implementación de la solución

Para la implementación del *framework* se ha definido las siguientes fases como parte del Ciclo de Vida de los Servicios ASO. En la implementación se describe las precondiciones, los actores que intervienen en la fase y sus respectivas tareas, resultados o salidas y finalmente algunas consideraciones a tomar en cuenta.

Diseño de Servicios Web REST:

Fase inicial, en el que el Diseñador asignado a un proyecto realiza un levantamiento de información acerca de las necesidades concernientes a los Servicios Web REST.

Tabla 9: Flujo del Proceso de Diseño.
Fuente: Elaboración propia.

Precondiciones:	Estado Ticket Jira: “In Progress”.
Tareas:	Diseñador: Trabaja con el equipo SCRUM para diseñar la solución que cumpla los requerimientos del proyecto. Definir el alcance que tendrá o tendrán los servicios REST.
Resultados:	Documentos completados y publicados: - Ticket Jira (PAA7) - D520 - Ficha técnica de <i>Backend</i>. - SMC - Ficha Técnica de Servicios ASO. - Descriptores.

Solicitud de Información:

Es importante para los Equipos Scrum contar con la participación del Gestor de Servicios en fases muy tempranas. En esta fase el Equipo Scrum podrá responder las siguientes preguntas del Gestor asignado:

1. ¿Cuántos y cuáles son los Servicios REST que abarcará el proyecto?
2. ¿Cuáles son los alcances de cada Servicio REST?
3. ¿Cuántos y cuáles canales conforman el proyecto?
4. ¿Está decidido pasar a Producción?
5. ¿Los servicios contienen campos que requieren cifrado?
6. ¿el *backend* que el servicio consumirá es interno o externo al banco? ¿se requiere alta de proxy y/o instalación de certificados?

Tabla 10: Flujo del Proceso de Solicitud de Información.

Fuente: Elaboración propia.

Tareas:	Gestor de Servicio: A inicios de cada PI, envía un correo al Equipo SCRUM solicitando información sobre el proyecto y los servicios que se atenderán como parte de este. Equipo SCRUM: Entrega la información requerida, dando respuesta al correo enviado por el Gestor de Servicio.
Resultados:	Información compartida, clara y precisa. MVPs asociados al proyecto. Información de integrantes del equipo para coordinaciones. Lista de servicios que se atenderán en el presente PI: - Información del servicio: RegistryId y LogicalId. - Tipo de gestión: Nuevo, Modificación, Reuso. - Datos de prueba e Información de <i>backend</i> : URL, Proxy, Certificados.

Construcción:

En esta fase se ha finalizado el diseño del Servicio REST.

Tabla 11: Flujo del proceso de Construcción.
Fuente: Elaboración propia.

Precondiciones:	Diseño completado. Estado Ticket Jira: “ToRework”.
Tareas:	Fábrica de Desarrollo (SLA de 7 días): Desarrolla los servicios nuevos o modificados de los servicios en las API’s pertenecientes Equipo SCRUM: Confirma que el entorno <i>backend</i> Desarrollo y/o Test está disponible Gestor de Servicio: Realiza el despliegue de los componentes en Jenkins.
Resultados:	API’s desplegadas en los entornos de Desarrollo y/o Test
Consideraciones:	El Equipo scrum debe considerar los siguientes lineamientos de desarrollo: Todos los requerimientos de nuevas APIs se implementarán con el bloque de respuesta del error en modo GABI.

Configuración:

Para el correcto funcionamiento de los Servicios REST es importante ejecutar algunas configuraciones externas al servicio, tales como son: cifrado, catalogación, alta de enumerativos, Gestión de Errores, alta de proxy e instalación de certificados.

Tabla 12: Flujo del Proceso de Configuración.
Fuente: Elaboración propia.

Precondiciones:	Diseño completado. Servicio y Canal catalogado por Seguridad. Estado Ticket Jira: “To Rework”.
Tareas:	Diseñador: Informa al Gestor de Servicio sobre propiedades, gestión de errores, <i>enums</i> y llaves de cifrado a utilizar. Equipo SCRUM: Entrega al Gestor de Servicio información de <i>backend</i> (URL, Proxy, Certificados). Entrega al Gestor de Servicio datos de prueba (de 3 casos por servicio). Gestor de Servicio (SLA de 7 días): Configura los parámetros del servicio, así como la comunicación con el <i>backend</i> .
Resultados:	- Propiedades del servicio configuradas. - Llaves de cifrado configuradas. - Gestión de errores configurado. - Enumerativos configurados. - <i>Backend</i> configurado (URL, Proxy, Certificados).

Test Arquitectura:

En esta fase se ejecutan las pruebas funcionales obedeciendo al principio de la detección temprana de fallas.

Tabla 13: Flujo del Proceso de Test de Arquitectura.
Fuente: Elaboración propia.

Precondiciones:	Implementación completada. Configuraciones finalizadas. <i>Backend</i> disponible en el entorno de Test. Datos de prueba enviados por el Equipo SCRUM. Estado Ticket Jira: “Test”.
Tareas:	Gestor de Servicio (SLA de 7 días): Prueba los servicios solicitados con los datos de prueba facilitados por el Equipo SCRUM : Identifica y reporta problemas / incidencias sobre sus pruebas. Documenta las pruebas realizadas. Acompaña y resuelve las dudas del Gestor de Servicio durante las pruebas. Fábrica de Desarrollo : Corrige las incidencias reportadas por el Gestor de Servicio.
Resultados:	Conformidad de las pruebas por parte del Gestor de Servicio. Documentación completada.

Test Aplicativo:

Fase en la que el Gestor de Servicios REST ejecuta pruebas haciendo uso de los Datos de Pruebas proporcionados por el Equipo Scrum.

Tabla 14: Flujo del Proceso de Test Aplicativo.
Fuente: Elaboración propia

Precondiciones:	Documento T907 en el entorno de Test completado. Pruebas con <i>backend</i> satisfactorias (flujos básicos). Estado Ticket Jira: “Ready to Deploy”.
Tareas:	Equipo SCRUM: Prueba y valida los servicios en el entorno de Test. De ser el caso, reporta las incidencias identificadas en sus pruebas. Al finalizar las pruebas, comunica por correo al Gestor de Servicio. Confirmación que el <i>backend</i> en el entorno de Calidad está disponible. Confirmación de pruebas satisfactorias en el entorno de Test con evidencia. Solicitud del despliegue de los cambios al entorno de Calidad. Gestor de Servicio: Atiende, analiza y determina las incidencias presentadas durante este tiempo. Acompaña y resuelve las dudas del Equipo SCRUM durante sus pruebas. Coordina y realiza el despliegue de los componentes al entorno de Calidad. Fábrica de Desarrollo: Corrige las incidencias reportadas por el Gestor de Servicio.
Resultados:	Componentes desplegados en entorno de Calidad

Calidad Aplicative:

En esta Fase el Equipo Scrum ejecuta el Plan de Pruebas.

Tabla 15: Flujo del Proceso de Calidad Aplicative.

Fuente: Elaboración propia.

Precondiciones:	Pruebas en el entorno de Test concluidas. <i>Backend</i> disponible en el entorno de Calidad. Componentes desplegados en el entorno de Calidad. Estado Ticket Jira: “Accepted”.
Tareas:	Equipo SCRUM: Certifica el correcto funcionamiento de los servicios. De ser el caso, reporta las incidencias identificadas en sus pruebas. Confirma por correo al Gestor de Servicio la certificación de los servicios y solicita la gestión de los componentes integrados para Pruebas de no impacto. De ser el caso, realiza pruebas preliminares en el entorno de Test con los componentes integrados para asegurar su correcta configuración. Coordina las Pruebas de no impacto por correo con la siguiente información: Lista de APIs modificadas y los Canales impactados. Fecha y Hora de la ventana de pruebas coordinada con los Canales impactados. Nota: La duración de la ventana es de 2 horas como máximo y sólo puede programarse en las mañanas (hasta las 2pm) para mantener el entorno disponible para certificaciones en las tardes.
Tareas:	Equipo SCRUM: Durante las Pruebas de no impacto: En caso de incidencias, coordina con el Gestor de Servicio para que identifique la causa y resuelva lo reportado. Coordina y realiza el seguimiento de los Canales impactados para asegurar su conformidad. Nota: Es responsabilidad del Equipo SCRUM obtener la conformidad de los Canales impactados. De faltar alguna no se podrá avanzar con la gestión del pase a producción. Al cierre de las Pruebas de no Impacto (Sin observaciones pendientes y Conformidad de todos los Canales impactados) Comunica al Gestor de Servicio y solicita iniciar con la gestión del pase a producción. Registra la Solicitud de pase a producción en el Formulario (proporcionado por Gestión de Servicios). Luego de comunicar y registrar la solicitud, programa la fecha de pase del pase en el comité de pases con ASO y Gestión de Cambios.

Tabla 16: Flujo del Proceso de Calidad Aplicative (continuación)

Fuente: Elaboración propia.

Tareas:	Gestor de Servicio: Atiende, analiza y determina las incidencias presentadas durante este tiempo. Acompaña y resuelve las dudas del Equipo SCRUM durante sus pruebas. Envía la lista de Canales impactados al Equipo SCRUM para que coordine la ventana de pruebas. Solicita al equipo de Arquitectura de Servicios los componentes integrados para la ventana de pruebas. De ser el caso, y con el apoyo del Equipo SCRUM, prepara una ventana de pruebas preliminar en el entorno de Test con los componentes integrados para asegurar su correcta configuración. Realiza el seguimiento de la ventana de pruebas de no impacto para responder lo antes posible cualquier observación o incidencia de los Canales impactados. Fábrica de Desarrollo: Corrige las incidencias reportadas por el Gestor de Servicio.
Resultados:	Solicitud a pase a producción enviada

Preparación para pase a Producción:

Esta fase es crucial ya que se evidenciará si todo el trabajo realizado en fases anteriores fue realizado de manera correcta o no.

Tabla 17: Flujo del Proceso de Pase a Producción.

Fuente: Elaboración propia.

Precondiciones:	Fecha de pase programada en el comité y autorizada por el equipo de Gestión de Cambios.
Tareas:	Gestor de Servicio: Crea las Hojas P110 y Ticket <i>release</i> JIRA Realiza la gestión para la configuración del entorno Productivo (<i>backend</i>, propiedades, entre otros). Prepara los componentes para su despliegue al entorno de Producción. Equipo SCRUM: Entrega al Gestor de Servicio información del <i>backend</i> (URL, Certificados, Proxy) Gestiona las firmas de las hojas P110. Coordina con los responsables de los Canales impactados para las pruebas de no impacto en el horario del pase a producción. Coordina con el <i>Quality Coach</i> para su visto bueno en el Ticket <i>Release</i>. Nota: Se debe contar con la documentación y los visto bueno necesarios 2 días antes de la fecha del pase para que el equipo de Gestión de Cambios autorice el pase.
Resultados:	Cambios autorizados. Gestiones para el pase completadas.

Pase a Producción:

Ejecución del Plan de Pase a Producción. En esta fase participan equipos transversales locales e internacionales. Se realiza un trabajo en conjunto para pasar un cambio a producción sin poner en riesgo la estabilidad y la alta disponibilidad de los Servicios REST.

Tabla 18: Flujo del Proceso de Pase a Producción.
Fuente: Elaboración propia.

Precondiciones:	Cambios autorizados. Componentes listos para su despliegue al entorno de Producción.
Tareas:	Gestor de Servicio: Envía un correo a los involucrados en el pase para las pruebas de no impacto en los entornos de Liga Oculta y Producción. En caso no se cuente con el visto bueno de todos los canales impactados al finalizar el pase, revierte los cambios desplegados en el entorno de Producción. Realiza el seguimiento de las pruebas de no impacto. Equipo SCRUM: En caso de incidencias, coordina con el Gestor de Servicio para que identifique la causa y atienda lo reportado. Nota: Si el problema depende del equipo SCRUM, también deben participar de las coordinaciones con Frente Único. Coordina y realiza el seguimiento de los Canales impactados para asegurar su conformidad. Nota: Es responsabilidad del Equipo SCRUM obtener la conformidad de los canales impactados. De faltar alguna finalizar la ventana de pruebas del pase, el Gestor de Servicio procede a realizar el <i>rollback</i> del despliegue.
Resultados:	Despliegue de componentes completado.

Post pase a Producción:

Fase final en la que el Gestor deberá monitorear los Servicios REST desplegados en ambientes productivos.

Tabla 19: Flujo del Proceso de Post Pase a Producción.

Fuente: Elaboración propia

Precondiciones: Pase a producción completado satisfactoriamente.	
Tareas:	Equipo SCRUM: En caso de incidencias, reporta el caso al Gestor de Servicio para que priorice su revisión. Nota: En general, las incidencias productivas deben ser reportadas al equipo de Arquitectura de Servicios (ellos pueden derivarla al Gestor de Servicio). La atención directa por el Gestor de Servicio sólo corresponde en caso la incidencia esté vinculada a un pase a producción realizado recientemente (hasta una semana antes). Gestor de Servicio: Analiza y atiende de ser necesario las incidencias reportadas.
Resultados:	Despliegue a producción completado y estable.

3.3 Evaluación

Se obtuvieron datos concernientes a los indicadores previos a la implementación del *Framework*. Dichos indicadores muestran en un periodo de tiempo de 6 meses la cantidad de Pases a Producción fallidos. Estas cifras muestran, en principio, la poca o nula capacidad del equipo de auto gestionarse para poder llevar con éxito cambios en producción.

El objetivo de la Gestión de Servicios REST se centra en llevar cambios a producción que respondan a los requerimientos de los Equipos SCRUM.

Se visualiza en la Figura 19 lo siguiente:

1. En el periodo de 6 meses se llevaron a cabo 187 pases a producción de los cuales 99 fueron fallidos.
2. Las causas principales de pases a producción fallidos se deben a:
 - a. Defectos de fábrica de los servicios REST (siendo así 47 o 27.3%)
 - b. Mala configuración en los servicios REST (siendo así 17 o 9.9%)

Es aquí donde surgen las primeras banderas rojas que indicaba una problemática que requieren un análisis de la causa raíz.

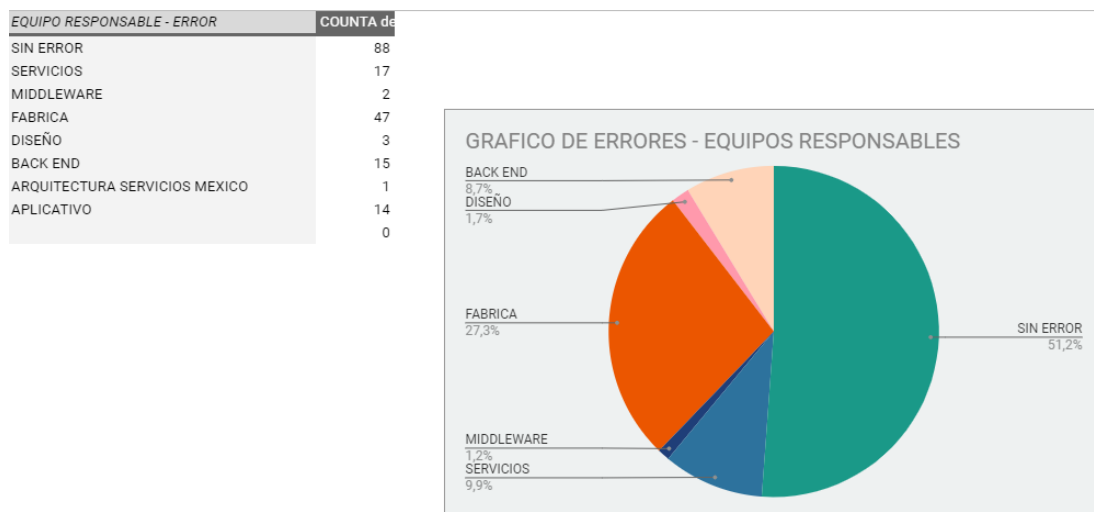


Figura 21 : Pases a Producción – Periodo: enero 2019 a junio 2019.
Fuente: Elaboración propia.

Los defectos de fábrica más conocidos son:

- ✓ Errores en los descriptores de los servicios.
- ✓ Desarrollo inconsistente a la solicitud técnica o requerimiento del mismo.
- ✓ Mapeo de campos (nulidad, obligatoriedad, longitud de campo, tipo de campo, etc.)
- ✓ Campos y llaves en duro (no mapeados por la arquitectura)

Defectos más usuales en la configuración de Servicios REST son:

- ✓ Configuraciones de Reuso de Servicios o Catalogación.
- ✓ Configuración de Cifrado.
- ✓ Alta de URL *Backend*.
- ✓ Alta de Enumerativos.
- ✓ Gestión de Errores.
- ✓ Alta de Proxy.
- ✓ Instalación de Certificado.
- ✓ Propiedades Externas.

Estas configuraciones tienen un alto impacto en el buen funcionamiento de los Servicios REST.

Posteriormente a la implementación del framework se obtuvieron los siguientes resultados (ver Figura 20) Para ellos se realizó un levantamiento de información considerando la misma cantidad de tiempo, 6 meses comprendidos desde el mes de enero del año 2020 hasta junio del mismo año. Tener en cuenta que se tomaron dos periodos que inician en los mismos meses, esto es importante ya que la cantidad de pases a producción varían en el año.

Se visualiza en la Figura 20 y 21 lo siguiente:

1. En dicho periodo de tiempo se llevaron a cabo 179 pases a producción, de los cuales 132 fueron exitosos lo que representa el 79.5%.
2. De los 47 pases a producción fallidos se puede visualizar que la causa principal de rollback es por fallas relacionadas al *backend*.
3. De todas las causas de rollback, se visualiza que sólo el 18.4% y el 10.5% están relacionados por fallas de Servicios y Diseño respectivamente (ver figura 21)

Se obtiene, de este modo, lo que el equipo de Gestión de Servicios REST deseaba alcanzar:

1. Reducción en los índices de pases a producción fallidas.
2. Reducción de la cantidad de fallas relacionadas a la Gestión de Servicios y Diseño de Servicios REST.

Es preciso mencionar que la reducción en el índice de pases a producción se debe a todo el trabajo que se realiza a lo largo del Ciclo de Vida de los Servicios REST, a una correcta gestión de estos servicios y al aseguramiento de la calidad en términos de procesos, roles y tiempos de atención.

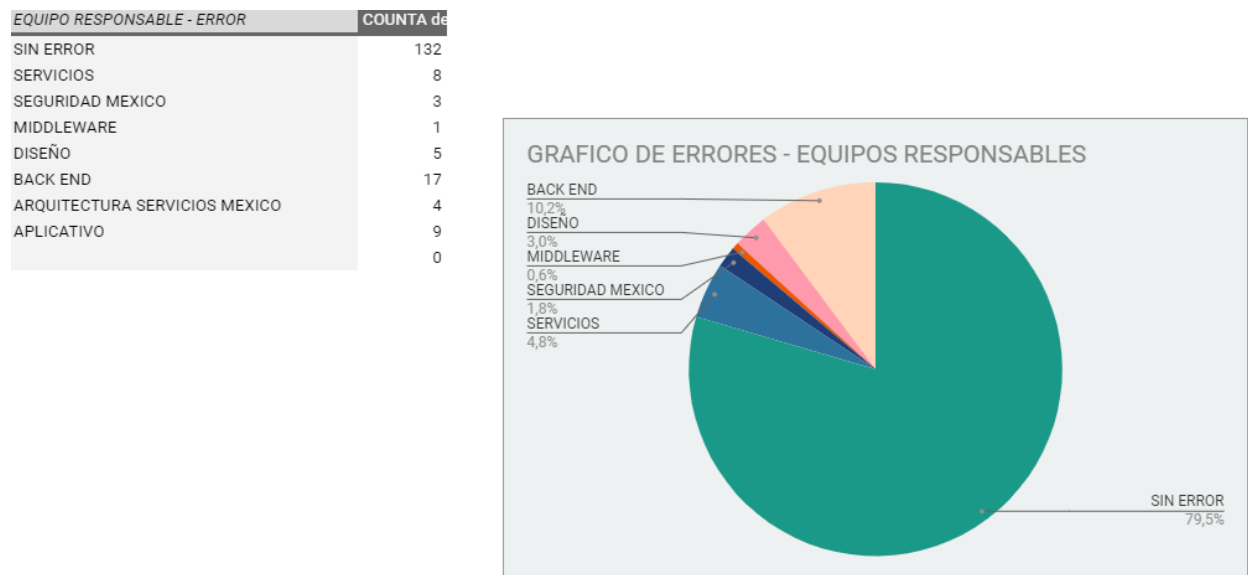


Figura 22: Pases a Producción – periodo enero 2020 a junio 2020.
Fuente: Elaboración propia.

EQUIPO RESPONSABLE - ERROR	COUNTA de
SERVICIOS	7
MIDDLEWARE	1
DISEÑO	4
BACK END	17
ARQUITECTURA SERVICIOS MEXICO	2
APLICATIVO	7
	0

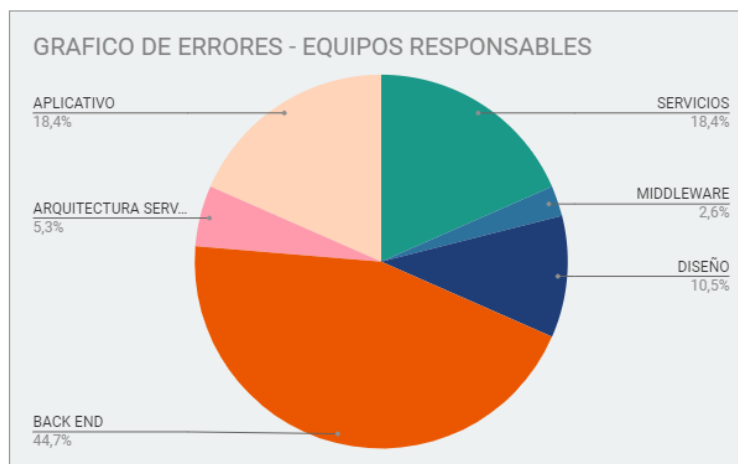


Figura 23: Motivos de Rollback – periodo enero 2020 a junio 2020.

Fuente: Elaboración propia.

De este modo, se obtienen los siguientes beneficios:

1. Un *framework* diseñado específicamente para atender las necesidades del área de Arquitectura de Servicios de una entidad bancaria y totalmente ajustable a futuros requerimientos.
2. Los acuerdos del nivel de servicio requeridos para asegurar la calidad del mismo que brinda la empresa Entelgy a la entidad bancaria.
3. Definición de roles, flujos y tiempos de atención que son muy importantes para cumplir con el ANS a la par que se van afinando según lo requiera la gerencia.
4. Un *framework* que tiene como propósito alcanzar los objetivos del área en cuanto a productividad de cada gestor.
5. Un *framework* que proporcione la capacidad del área y que ayude a planificar y determinar si el área cuenta con los recursos de personas de acuerdo con sus conocimientos técnicos y habilidades blandas. De este modo, el área sabrá cuántos proyectos puede aceptar por cada PI.

Capítulo IV

Reflexión Crítica de la Experiencia

Aporte en el área de desarrollo y responsabilidades

El autor del presente trabajo ha tenido una participación en el análisis situacional, diseño, implementación y mejora del marco de trabajo que en este documento se expone. Dicha participación ha sido activa y medular, llevando a cabo sus conocimientos adquiridos en sus estudios y experiencias anteriores.

Es vital reconocer la apertura y disponibilidad otorgada por el líder técnico del equipo, la misma que hizo posible la atención del problema y la iniciativa de poner en marcha este trabajo.

Se realizó un análisis detallado a fin de poner sobre la mesa los puntos de dolor tal como se detalla en el capítulo III de este trabajo. En seguida, se definieron los diferentes flujos ya existentes, esto nos permitió ahorrar tiempo ya que decidimos analizar el funcionamiento del flujo y resaltar aspectos positivos que deberían permanecer en él.

La mentalidad del equipo siempre fue mejorar y no replantear, se reconoció que esta práctica es ideal para resolver problemáticas cuyas soluciones se esperan que sea llevado en el menor tiempo posible.

Teniendo identificados y afinados los flujos se prosiguió a la puesta en marcha teniendo en cuenta que como área podíamos decidir cuáles eran nuestros flujos y de qué manera atenderíamos los diferentes requerimientos de otras arquitecturas. El refinamiento de este *framework* se realizó gracias a las reuniones semanales que nos ayudaron a identificar y prevenir problemas potenciales.

Capítulo V

Conclusiones y Recomendaciones

Es de suma importancia reconocer que estamos en constante cambio y que lo que funciona el día de hoy, no funcionará con la misma exactitud y excelencia el día de mañana. Este trabajo detalla en esencia cómo el diseño y la implementación de un *framework* para la mejora en la gestión de servicios REST, ha podido cubrir la necesidad de resolver los problemas más relevantes que se dieron en un momento determinado.

Es importante el trabajo de refinamiento del flujo ya que esto no has ayudado a poder corregir de manera proactiva las falencias en el *framework*.

1.1 Conclusiones

El análisis realizado, el diseño y la implementación que se detalla en este trabajo dan paso a las siguientes conclusiones:

1. Realizar un análisis holístico de la situación previa a la implementación ha sido de gran ayuda para la detección de las causas de los problemas descritos en este documento.
2. El concepto de mejora continua a través de reuniones de retrospectiva ha sido muy efectivo gracias a la iniciativa y participación de cada uno de los integrantes.
3. El hecho de no replantear los flujos existentes sino de reciclar ha sido muy efectivo en pro del tiempo utilizado en el diseño de la solución.
4. Los cambios son realizados gracias al comportamiento de las personas que lo ejecutan. Las herramientas y las tecnologías utilizadas para llevar a cabo esos cambios pasan a segundo plano. Dicho de esta manera las metodologías, las buenas prácticas y las herramientas no debería ser un logro, más bien el principal reto es generar la colaboración entre los integrantes del equipo.
5. El *framework* tiene como base la flexibilidad y la capacidad de ser un sistema vivo que está en constante cambio a la medida que los roles y flujos pasan por un proceso de mejora continua.

1.2 Recomendaciones:

1. Se recomienda no perder el objetivo por el cual nace este *framework*. En términos generales el objetivo más importante es de brindar un mejor servicio como equipo contratado. Ya que con la experiencia esto nos ha demostrado que abre nuevas oportunidades de negocio y fortalece la confianza entre empresa y cliente.
2. Se recomienda no subestimar cada fase de este *framework* ya que los procesos, funciones, roles y tiempos de atención podrían cambiar en cualquier momento, sin embargo, los principios sobre los cuales está basado son inalterables.
3. El éxito que se obtenga en la Gestión de Servicios REST del Banco dependerá en gran medida de la buena aplicación que se dé al *framework*.
4. Lo más importante es mantener el modelo actualizado y ajustable a las nuevas necesidades del cliente debido a los cambios emergentes.

Bibliografía

- blog.bi-geek.com. (19 de Junio de 2018). *Servicios Web: RESTful*. Obtenido de <https://blog.bi-geek.com/servicios-web-restful/>
- De Seta, Leonardo. (13 de Noviembre de 2018). *Introducción a los servicios web RESTful*. Obtenido de <https://dosideas.com/noticias/java/314-introduccion-a-los-servicios-web-restful>
- Díaz, J. (2020). *Plan para implantación de un "testing"*. Alicante - España: Posgrado Universidad de Alicante. Obtenido de <http://hdl.handle.net/10045/109760>
- ENTELGY - SALA DE PRENSA. (7 de Abril de 2021). *Los 6 retos a los que se enfrentan los servicios de Outsourcing*. Obtenido de Sala de Prensa: <https://www.entelgy.com/sala-de-prensa-k2/sala-de-prensa/los-6-retos-a-los-que-se-enfrentan-los-servicios-de-outsourcing>
- ENTELGY. (2021). *Divisiones de Entelgy*. Obtenido de <https://www.entelgy.com/divisiones>
- ENTELGY. (2021). *Entelgy: somos diferenciales*. Obtenido de [www.entelgy.com: https://www.entelgy.com/entelgy](https://www.entelgy.com/entelgy)
- ENTELGY SALA DE PRENSA. (7 de Julio de 2021). Obtenido de Metodología Agile: las claves para manejar equipos outsourcing con éxito: <https://www.entelgy.com/sala-de-prensa-k2/sala-de-prensa/metodologia-agile-las-claves-para-manejar-equipos-outsourcing-con-exito>
- EVALUANDO SOFTWARE. (29 de Junio de 2018). *¿Qué es la metodología 'Agile'?* Obtenido de <https://www.evaluandosoftware.com/la-metodologia-agile/>
- IT Process Maps. (2021). Obtenido de ITIL Gestion del Nivel de Servicio - SLM: https://wiki.es.it-processmaps.com/index.php/ITIL_Gestion_del_Nivel_de_Servicio_-_SLM
- KANBANIZE. (2020). *Qué es Kanban: Definición, Características y Ventajas*. Obtenido de <https://kanbanize.com/es/recursos-de-kanban/primeros-pasos/que-es-kanban>

- Mejía-Granda, C. (2018). *Desarrollo de servicios web REST “inseguros” para auto-aprendizaje en la explotación de vulnerabilidades*. Quito: Universidad Internacional de La Rioja.
- Mera Paz, J. M. (2017). *Análisis sistemático de información de la Norma ISO 25010 como base para la implementación en un laboratorio de Testing de software en la Universidad Cooperativa de Colombia Sede Popayán*. Cauca: Universidad Cooperativa de Colombia.
- Puerta, J. (2015). *Desarrollo de una API para la descripción y gestión de Servicios Web REST*. Castelló: Universitat Jaume I. Obtenido de <http://hdl.handle.net/10234/156006>
- RED HAT. (2021). *Qué son las API y para qué sirven*. Obtenido de <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>
- SCRUM GUIDE. (Noviembre de 2020). *La Guía SCRUM*. Obtenido de <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf>
- solocodigoweb.com. (15 de Setiembre de 2016). Obtenido de Cuando usar SOAP y cuando usar REST: <https://solocodigoweb.com/blog/2016/09/15/cuando-usar-soap-cuando-usar-rest/>

Anexos

1. Actividades por fases de cada tipo de requerimiento según su alcance: Reuso de Servicios.

Tabla 20: Consideraciones para el Reuso de Servicios.

Fuente: Elaboración propia.

ALCANCE: REUSO DE SERVICIOS REST:			
Escenario: Equipo Scrum requiere que un canal consuma un servicio que ya existe. Se requiere configuraciones de seguridad.			
Status JIRA	Fase	ANS	Tareas y dependencias
NEW	Inicio	NA	* Diseño crea ticket jira
IN PROGRESS	Diseño	NA	* Solicita catalogación de servicio. * Diseño asigna ticket Jira a Gestor de Servicios. * Diseño deja el ticket jira en el <i>stage</i> "Listo para construcción"
READY		8 horas	* En este <i>stage</i> se encuentran los tickets asignados a fabrica y a los Gestor de Servicio. * Cuando fabrica o el Gestor de Servicio envíe el ticket al <i>stage</i> "Construcción", este cambio indica que recién fabrica o Gestor de Servicios está atendiendo el requerimiento.
TO REWORK	Construcción	32 horas	* Registrar propiedades de cifrado del servicio en caso lo requiera. * Registrar enumerativos, gestión de errores y propiedades en caso lo requiera. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al <i>stage</i> "bloqueado". * Configuración de rutinas externas.
TEST	Test Arquitectura	24 horas	* Validar catalogación del servicio. * Validar cifrado del servicio en caso lo requiera. * Validar enumerativos, gestión de errores y propiedades. * Pruebas del servicio. Probar si está catalogado y si está haciendo el cifrado. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al <i>stage</i> "bloqueado". * Elaborar T907

Tabla 21: Consideraciones para el Reuso de Servicios (continuación)

Fuente: Elaboración propia.

ALCANCE: REUSO DE SERVICIOS REST:			
Escenario: Equipo Scrum requiere que un canal consuma un servicio que ya existe. Se requiere configuraciones de seguridad.			
Status JIRA	Fase	ANS	Tareas y dependencias
READY TO DEPLOY	Test Aplicativo	NA	* Gestor de Servicios solicita Datos a Aplicativo (datos externos al servicio como por ejemplo url back, etc.) datos del entorno de calidad. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el GS debe enviar el ticket al <i>stage</i> "bloqueado".
ACCEPTED	Calidad Aplicativo	NA	* Posterior a las pruebas funcionales de aplicativo en el entorno de calidad, el Gestor de Servicios solicita a Aplicativo que envíe el requerimiento de pruebas de no impacto a equipo de operaciones. * GS da el soporte Pruebas de no impacto * Con el visto bueno de las pruebas de no impacto, Aplicativo debe llenar el formulario de pases a producción. * Para el caso de reuso, las pruebas de no impacto no requieren pruebas de otros canales.
DEPLOYED	Producción	NA	* Posterior al pase productivo, Gestión de Cambios debe enviar el ticket <i>release</i> a la etapa de "Producción". * Gestor de Servicios debe enviar los tickets jira de servicios que pasaron a producción a la etapa de "DEPLOYED".
BLOCKED	Bloqueado	40 horas	* Ticket con status "bloqueado" por más de 5 días, debe pasar a status "descartado"
DISCARDED	Descartado	NA	* Servicios que fueron descartados y soliciten su reactivación, deben ser a través de un nuevo ticket jira.

2. Actividades por fases de cada tipo de requerimiento según su alcance: Modificación de Servicios.

Tabla 22: Consideraciones para Modificación de Servicios REST.

Fuente: Elaboración propia

ALCANCE MODIFICACIÓN DE SERVICIOS REST:			
Escenario: Equipo Scrum requiere la modificación de un servicio a nivel de código o de alguna configuración externa. Esta solicitud se realiza como mejora o evolutivo menor del servicio sin impactar su funcionalidad.			
Status JIRA	Fase	ANS	Tareas y dependencias
NEW	Inicio	ANS	* Diseño crea ticket jira
IN PROGRESS	Diseño	24 horas	* Solicita catalogación de servicio. * Diseño asigna ticket Jira a Fabrica * Diseño deja el ticket jira en el stage de "Listo para construcción" * Diseño debe generar una sub tareas para configuraciones de servicio (enumerativos, certificados, propiedades de cifrado, url, etc.)
READY (SLA 1 día)		8 horas	* En este stage se encuentran los tickets asignados a fabrica y a los Gestor de Servicios. * Cuando fábrica o el Gestor de Servicios envíe el ticket al stage "Construcción", este cambio indica que recién fabrica o Gestor de Servicios está atendiendo el requerimiento.
TO REWORK (4 días)	Construcción	32 horas	* Validar configuraciones. * Gestor de Servicios solicita datos de prueba a aplicativo * Fabrica inicia la construcción del servicio. * Fabrica envía componentes a Gestor de Servicios y envía el ticket al stage "Test Arquitectura" * Fabrica asigna el ticket al Gestor de Servicios. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al stage "bloqueado". * Configuración de rutinas externas.

Tabla 23: Consideraciones para Modificación de Servicios REST (continuación)

Fuente: Elaboración propia.

ALCANCE MODIFICACIÓN DE SERVICIOS REST:			
Escenario: Equipo Scrum requiere la modificación de un servicio a nivel de código o de alguna configuración externa. Esta solicitud se realiza como mejora o evolutivo menor del servicio sin impactar su funcionalidad.			
Status JIRA	Fase	ANS	Tareas y dependencias
TEST (3 días)	Test Arquitectura	24 horas	* Validar catalogación del servicio. * Validar cifrado del servicio en caso lo requiera. * Validar enumerativos, gestión de errores y propiedades. * Pruebas del servicio. Probar si está catalogado y si está haciendo el cifrado. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al stage "bloqueado". * Elaborar T907
READY TO DEPLOY	Test Aplicativo	NA	* Gestor de Servicios solicita Datos a Aplicativo (datos externos al servicio como por ejemplo url back, etc.) datos del entorno de calidad. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al stage "bloqueado".
ACCEPTED	Calidad Aplicativo	NA	* Posterior a las pruebas funcionales de aplicativo en el entorno de calidad, el Gestor de Servicios solicita a Aplicativo que envíe el requerimiento de pruebas de no impacto a equipo de operaciones. * Gestor de Servicios da el soporte Pruebas de no impacto * Con el Visto Bueno de las pruebas de no impacto, Aplicativo debe llenar el formulario de pases a producción. * Gestor de Servicios envía canales impactados al solicitante de Pruebas de No Impacto.
DEPLOYED	Producción	NA	* Posterior al pase productivo, Gestión de Cambios debe enviar el ticket release a la etapa de "Producción". * Gestor de Servicios debe enviar los tickets jira de servicios que pasaron a producción a la etapa de "DEPLOYED".
BLOCKED	Bloqueado	40 horas	* Ticket con status "bloqueado" por más de 5 días, debe pasar a status "descartado"
DISCARDED	Descartado	NA	* Servicios que fueron descartados y soliciten su reactivación, deben ser a través de un nuevo ticket jira.

3. Actividades por fases de cada tipo de requerimiento según su alcance: Creación de Servicios.

Tabla 24: Consideraciones para la creación de Servicios REST.

Fuente: Elaboración propia.

ALCANCE CREACIÓN DE SERVICIOS REST:			
Escenario: Equipo Scrum requiere la creación de un servicio nuevo, con especificaciones únicas a nivel de código y configuraciones externas al API. En este tipo de solicitud se puede desestimar las Pruebas de No Impacto.			
Status	Fase	ANS	Tareas y dependencias
JIRA			
NEW	Inicio	ANS	* Diseño crea ticket jira
IN PROGRESS	Diseño	32 horas	* Solicita catalogación de servicio. * Diseño asigna ticket Jira a Fabrica * Diseño deja el ticket jira en el stage de "Listo para construcción" * Diseño debe generar una sub tareas para configuraciones de servicio (enumerativos, certificados, propiedades de cifrado, url, etc.)
READY		8 horas	* En este stage se encuentran los tickets asignados a fabrica y a los Gestor de Servicios. * Cuando fabrica o el Gestor de Servicios envíe el ticket al stage "Construcción", este cambio indica que recién fabrica o Gestor de Servicios está atendiendo el requerimiento.
TO REWORK	Construcción	32 horas	* Validar configuraciones. * Gestor de Servicios solicita datos de prueba a aplicativo * Fabrica inicia la construcción del servicio. * Fabrica envía componentes a Gestor de Servicios y envía el ticket al stage "Test Arquitectura" * Fabrica asigna el ticket al Gestor de Servicios. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al stage "bloqueado". * Configuración de rutinas externas. * En caso la ficha técnica corresponda a un Canal Nuevo, debemos validar que ya se hayan realizado pruebas del servicio que genera la llave de seguridad en los diferentes entornos, estas tareas son previas y realizadas por el equipo de operaciones.

Tabla 25: Consideraciones para la creación de Servicios REST (continuación)

Fuente: Elaboración propia

ALCANCE CREACIÓN DE SERVICIOS REST:			
Escenario: Equipo Scrum requiere la creación de un servicio nuevo, con especificaciones únicas a nivel de código y configuraciones externas al API. En este tipo de solicitud se puede desestimar las Pruebas de No Impacto.			
Status JIRA	Fase	ANS	Tareas y dependencias
TEST	Test Arquitectura	24 horas	* Validar catalogación del servicio. * Validar cifrado del servicio en caso lo requiera. * Validar enumerativos, gestión de errores y propiedades. * Pruebas del servicio. Probar si está catalogado y si está haciendo el cifrado. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al stage "bloqueado". * Elaborar T907: subirlos a Gestión de Servicios y en el directorio del equipo
READY TO DEPLOY	Test Aplicativo	NA	* Gestor de Servicios solicita Datos a Aplicativo (datos externos al servicio como por ejemplo url back, etc.) datos del entorno de calidad. * En caso de generarse alguna dependencia para la finalización de pruebas del equipo de arquitectura, inmediatamente el Gestor de Servicios debe enviar el ticket al stage "bloqueado".
ACCEPTED	Calidad Aplicativo	NA	* Posterior a las pruebas funcionales de aplicativo en el entorno de Calidad, el Gestor de Servicios solicita a Aplicativo que envíe el requerimiento de pruebas de no impacto a equipo de operaciones. * Gestor de Servicios da el soporte Pruebas de no impacto * Con el VoBo de las pruebas de no impacto, Aplicativo debe llenar el formulario de pases a producción. * Gestor de Servicios envía canales impactados al solicitante de Pruebas de No Impacto.
DEPLOYED	Producción	NA	* Posterior al pase productivo, Gestión de Cambios debe enviar el ticket release a la etapa de "Producción". * Gestor de Servicios debe enviar los tickets jira de servicios que pasaron a producción a la etapa de "DEPLOYED".
BLOCKED	Bloqueado	NA	* Ticket con status "bloqueado" por más de 5 días, debe pasar a status "descartado"
DISCARDED	Descartado	40 horas	* Servicios que fueron descartados y soliciten su reactivación, deben ser a través de un nuevo ticket jira.

4. Checklist y Mapeo de Servicios de un Equipo Scrum.

INFORMACION DEL SERVICIO				Configuraciones Externas [1]																Paquete Dimer	
tipo [2]	SN [3]	Logical ID (SMC) [4]	Ticket Jira [5]	AAP [6]	Mét.	Comentarios	Endpoint [7]	Postman	TEST OK?	Catalogado	Cifrados	URL Bknd [8]	Alta Proxy	Inst. Certif	Enums	Gest. Err	Proxy	Certificados	Prop Ext. [9]	Nombre Rut/Dep	tpl json [10]
N	SNPE1800089/businesses	listBusinessEconomicGroups	APG PAA7-1232	13000024/SFSC	GET	HOST	/businesses/	☐	☐	Pend	OK	OK	OK	OK	Pend	OK	OK	Pend	OK	_configuration-rm	☐
N	SNPE1800089/businesses	getBusinessEconomicGroups	APG PAA7-1233	13000024/SFSC	GET	HOST	/businesses/	☐	☐	OK	Pend	Pend	OK	OK	OK	OK	OK	OK	OK	processing-execu	☒
N	SNPE1900017/economic-groups	listEconomicGroups	APG PAA7-1458	13000024/SFSC	GET	HOST	/economic-g	☐	☐	Pend	OK	OK	Pend	Pend	Pend	Pend	Pend	Pend	Pend	ic_translator-rm.j	☐
N	SNPE1900017/economic-groups	listEconomicGroups	APG PAA7-1678	13000024/SFSC	GET	HOST	/economic-g	☐	☐	OK	OK	OK	OK	OK	Pend	OK	OK	Pend	OK	processing-execu	☒
N	SNGB19000011/RiskBusinessAdm	getEnginePath	APL PAA7-1442	13000024/SFSC	GET	REST	/risk-business	☐	☐	OK	Pend	Pend	Pend	Pend	Pend	Pend	Pend	Pend	Pend	ic_translator-rm.j	☒
N	SNGB19000011/RiskBusinessAdm	getEngineRatingCalc	APG PAA7-1444	13000024/SFSC	GET	REST	/risk-business	☐	☐	OK	OK	OK	OK	Pend	Pend	OK	Pend	Pend	OK	processing-execu	☒
N	SNGB19000011/RiskBusinessAdm	createRiskBusinessA	APG PAA7-1655	13000024/SFSC	POST	APX	/risk-business	☐	☐	OK	OK	Pend	NA	Pend	Pend	NA	Pend	Pend	NA	ic_translator-rm.j	☒
N	SNGB19000011/RiskBusinessAdm	getEngineFinancialSi	APG PAA7-1657	13000024/SFSC	GET	REST	/risk-business	☐	☐	Pend	OK	OK	OK	Pend	Pend	OK	Pend	Pend	OK	processing-execu	☒
N	SNGB19000021/ratings	getRatingQualitative	APG PAA7-1925	13000024/SFSC	GET	APX	/ratings/v1/q	☐	☐	Pend	OK	OK	OK	Pend	Pend	OK	Pend	Pend	OK	ic_translator-rm.j	☒

Figura 24: Checklist de Gestión para Pase a Producción de Servicios.

Fuente: Elaboración propia.

5. Flujo de Atención de Fallas en Ambientes Previos.

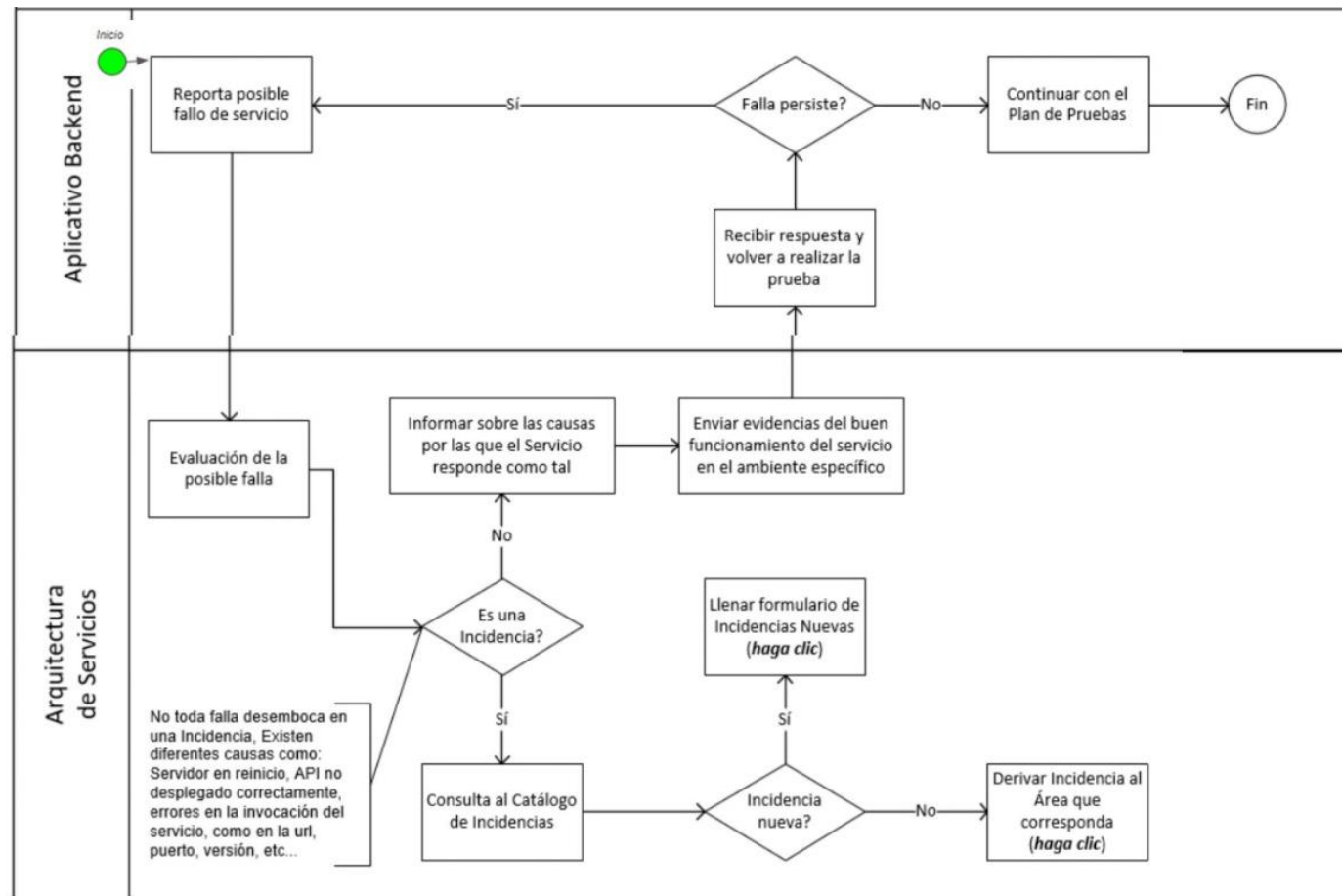


Figura 25: Flujo de Atención de Fallas en Ambientes Previos.
Fuente: Elaboración propia.

6. Flujo de Pases a Producción y ejecución de cambios en caliente.

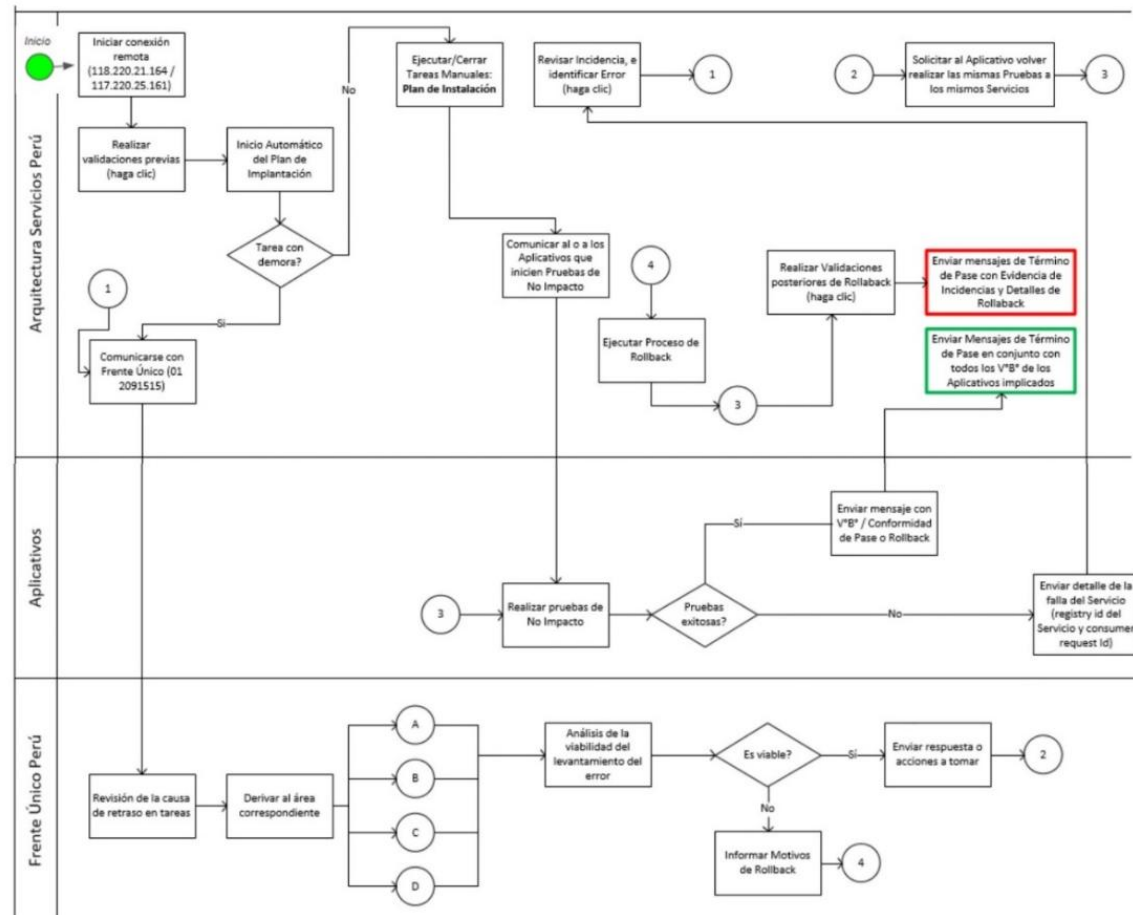


Figura 26: Flujo de Pases a Producción y ejecución de cambios en caliente.
Fuente: Elaboración propia.

7. Flujo de Pruebas Integrales de Servicios REST.

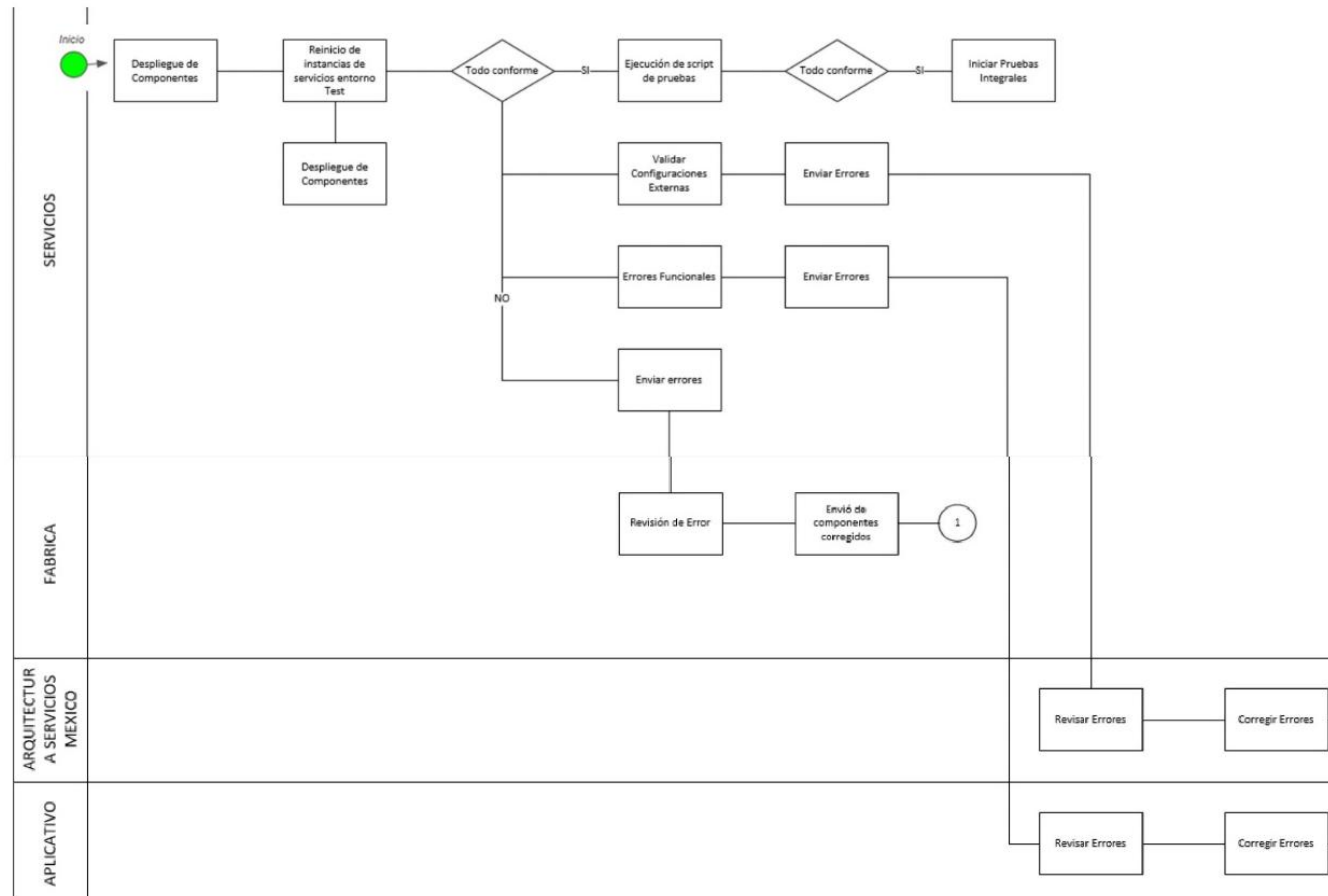


Figura 27: Flujo de Pruebas Integrales de Servicios REST.
Fuente: Elaboración propia