



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Software

Implementación de un sistema web usando la metodología Scrum para mejorar el proceso de capacitación y seguimiento a nuevos asesores de oficinas en una entidad financiera

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de Ingeniero de Software

AUTOR

Bryan Diego DELFIN ACHARTE

ASESOR

Mg. Augusto Parcemón CORTEZ VÁSQUEZ

Lima, Perú

2026



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

Delfin, B. (2026). *Implementación de un sistema web usando la metodología Scrum para mejorar el proceso de capacitación y seguimiento a nuevos asesores de oficinas en una entidad financiera*. [Trabajo de suficiencia profesional de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Software]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	Bryan Diego Delfin Acharte
Tipo de documento de identidad	DNI
Número de documento de identidad	47238575
URL de ORCID	https://orcid.org/0009-0009-7825-352X
Datos de asesor	
Nombres y apellidos	Augusto Parcemón Cortez Vásquez
Tipo de documento de identidad	DNI
Número de documento de identidad	08634618
URL de ORCID	https://orcid.org/0000-0002-5188-7962
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Jorge Raúl Díaz Muñante
Tipo de documento	DNI
Número de documento de identidad	07216161
Miembro del jurado 1	
Nombres y apellidos	César Augusto Alcántara Loayza
Tipo de documento	DNI
Número de documento de identidad	09132297
Datos de investigación	
Línea de investigación	No aplica
Grupo de investigación	No aplica
Agencia de financiamiento	Sin financiamiento

Ubicación geográfica de la investigación	País: Perú Departamento: Lima Provincia: Lima
	Distrito: Cercado de Lima Jr. Carlos Amezaga No. 375 Universidad Nacional Mayor de San Marcos Latitud: -12.0564232 Longitud: -77.0843327
Año o rango de años en que se realizó la investigación	2024
URL de disciplinas OCDE	2.02.04 -- 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 Software

Acta Virtual de Sustentación
del Trabajo de Suficiencia Profesional

Siendo las 20:03 horas del día 30 de marzo del año 2026, se reunieron virtualmente los docentes designados como Miembros de Jurado del Trabajo de Suficiencia Profesional, presidido por el Mg. Díaz Muñante Jorge Raúl (Presidente), Mg. Alcántara Loayza César Augusto (Miembro) y el Mg. Cortez Vásquez Augusto Parcemon (Miembro Asesor), usando la plataforma Meet (<http://meet.google.com/wcc-jfha-tms>), para la sustentación virtual del Trabajo de Suficiencia Profesional intitulado: **“IMPLEMENTACIÓN DE UN SISTEMA WEB USANDO LA METODOLOGÍA SCRUM PARA MEJORAR EL PROCESO DE CAPACITACIÓN Y SEGUIMIENTO A NUEVOS ASESORES DE OFICINAS EN UNA ENTIDAD FINANCIERA”**, por el Bachiller **Delfin Acharte Bryan Diego**; para obtener el Título Profesional de Ingeniero de Software.

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

El 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 habiendo efectuado la calificación correspondiente por los miembros del Jurado, el Bachiller obtuvo la nota de **17 (DIECISIETE)**.

A continuación el Presidente de Jurados el Mg. Díaz Muñante Jorge Raúl, declara al Bachiller **Ingeniero de Software**.

Siendo las 20:59 horas, se levantó la sesión.

Presidente

Mg. Díaz Muñante Jorge Raúl

Miembro

Mg. Alcántara Loayza César Augusto

Miembro Asesor

Mg. Cortez Vásquez Augusto Parcemon

CERTIFICADO DE SIMILITUD

Yo, Augusto Parcemon Cortez Vasquez en mi condición de asesor acreditado mediante Resolución Decanal N° 001268-2025-D-FISI/UNMSM y OFICIO N° 0013-PTTSP-FISI-2025 del Trabajo de Suficiencia Profesional, cuyo título es Implementación de un sistema web usando la metodología Scrum para mejorar el proceso de capacitación y seguimiento a nuevos asesores de oficinas en una entidad financiera, presentado por el bachiller Delfin Acharte Bryan Diego para optar el Título Profesional de Ingeniero de Software, CERTIFICO que se ha cumplido con lo establecido en la Directiva de Originalidad y de Similitud de Trabajos Académicos, de Investigación y Producción Intelectual. Según la revisión, análisis y evaluación mediante la plataforma PIDS, el documento evaluado cuenta con el 13% de similitud, nivel PERMITIDO para continuar con los trámites correspondientes y para su **publicación en el repositorio institucional**.

Se emite el presente certificado en cumplimiento de lo establecido en las normas vigentes, como uno de los requisitos para la obtención del **TITULO PROFESIONAL DE INGENIERO DE SOFTWARE**.

Lima, 13 de abril de 2026

Firma del Asesor:



Nombres y Apellidos del asesor: Augusto Parcemon Cortez Vasquez

DNI: 08634618

FICHA CATALOGRÁFICA

IMPLEMENTACIÓN DE UN SISTEMA WEB USANDO LA METODOLOGÍA SCRUM PARA MEJORAR EL PROCESO DE CAPACITACIÓN Y SEGUIMIENTO A NUEVOS ASESORES DE OFICINAS EN UNA ENTIDAD FINANCIERA

AUTOR: DELFIN ACHARTE, BRYAN DIEGO

ASESOR: CORTEZ VÁSQUEZ, AUGUSTO PARCEMON

LIMA – PERÚ, 2026

TÍTULO PROFESIONAL: Título Profesional de Ingeniero de Software

ÁREA / PROGRAMA / LÍNEA DE INVESTIGACIÓN:

Ingenierías/Tecnología de 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 Software

Formato 21 x 29.7 cm

Páginas: x, 62

DEDICATORIA

A mi esposa Sandra por su amor incondicional y su apoyo constante.

A mi hija Emilia por ser mi motivo para esforzarme cada día.

A mis padres Nancy e Hidelfredo por apoyarme en todo momento.

A toda mi familia en general que me brindaron siempre la confianza.

AGRADECIMIENTOS

A mi alma mater que gracias a su educación gratuita que me ayudó a poder conseguir mis objetivos personales y profesionales.

A mi asesor por su apoyo y paciencia que fue muy importante para poder desarrollar el presente trabajo.

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

**Implementación de un sistema web usando la metodología Scrum para mejorar el
proceso de capacitación y seguimiento a nuevos asesores de oficinas en una entidad
financiera**

Autor: Bryan Diego, Delfin Acharte
Asesor: Cortez Vasquez, Augusto Parcemon
Título: Trabajo de Suficiencia Profesional para optar el Título Profesional
de Ingeniero de Software
Fecha: Marzo 2026

RESUMEN

El presente trabajo de suficiencia profesional expone la implementación de un sistema web con el fin de mejorar el proceso de capacitación y seguimiento de nuevos asesores de servicios en una entidad financiera. El proyecto surge como respuesta a los desafíos significativos identificados en el área de formación, tales como la dispersión de la información, la falta de estandarización en los contenidos formativos y la alta carga operativa derivada de procesos manuales. El objetivo es implementar el sistema web para mejorar el proceso de formación y capacitación que lidera el área de formación. La solución fue gestionada bajo la metodología SCRUM. Se logró implementar el sistema web con los diferentes módulos para la gestión de usuarios, seguimiento de las evaluaciones, envíos de correos, reportes específicos, gestión de contenidos entre otros. El sistema web permitió mejorar el proceso de formación y esto impacta directamente en la experiencia del cliente, quien ahora recibe una atención más experta y resolutive, fundamentada en un proceso de formación robusto, medible y automatizado.

Palabras clave: Sistema Web, SCRUM, Capacitación, Automatización de Procesos

MAJOR NATIONAL UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEMS AND INFORMATICS ENGINEERING
PROFESSIONAL SCHOOL OF SOFTWARE ENGINEERING

**Implementation of a web-based system using the Scrum methodology to improve
the training and monitoring process for new office advisors at a financial
institution**

Author: Delfin Acharte, Bryan Diego

Advisor: Cortez Vásquez, Augusto Parcemon

Title: Professional Sufficiency Work to opt for the Professional Title of
Software Engineer

Date: March 2026

ABSTRACT

This professional proficiency project presents the implementation of a web-based system designed to improve the training and monitoring process for new service advisors at a financial institution. The project arose in response to significant challenges identified in the area of training, such as the dispersion of information, the lack of standardization in training content, and the high operational burden resulting from manual processes. The objective is to implement the web-based system to improve the training and education process led by the training department. The solution was managed using the SCRUM methodology. The web-based system was successfully implemented with different modules for user management, evaluation monitoring, emailing, specific reports, content management, among others. The web system improved the training process, which directly impacts the customer experience. Customers now receive more expert and responsive service based on a robust, measurable, and automated training process.

Key words: Web System, SCRUM, Training, Process Automation

TABLA DE CONTENIDO

RESUMEN	v
ABSTRACT	vi
TABLA DE CONTENIDO	vii
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS.....	x
INTRODUCCIÓN.....	1
CAPÍTULO I – TRAYECTORIA PROFESIONAL.....	3
1.1 PRESENTACIÓN PROFESIONAL	3
1.2 EXPERIENCIA PROFESIONAL	3
1.3 EDUCACIÓN Y CERTIFICACIONES.....	8
CAPÍTULO II - CONTEXTO EN EL QUE SE DESARROLLA LA EXPERIENCIA	10
2.1 EMPRESA – ACTIVIDAD QUE REALIZA	10
2.2 VISIÓN.....	10
2.3 MISIÓN	11
2.4 ORGANIZACIÓN DE LA EMPRESA	11
2.5 ÁREA, CARGO Y FUNCIONES DESEMPEÑADAS	12
2.6 EXPERIENCIA PROFESIONAL REALIZADA EN LA ORGANIZACIÓN	13
CAPÍTULO III - ACTIVIDADES DESARROLLADAS.....	15
3.1 ANTECEDENTES	15
3.2 SITUACIÓN PROBLEMÁTICA.....	16
3.2.1 DEFINICIÓN DEL PROBLEMA.....	18
3.3 SOLUCIÓN	19
3.3.1 OBJETIVOS.....	19
3.3.2 ALCANCE	20
3.3.3 ETAPAS Y METODOLOGÍA	23
3.3.4 FUNDAMENTOS UTILIZADOS	26
3.3.5 IMPLEMENTACIÓN DE LAS ÁREAS, PROCESOS, SISTEMAS Y BUENAS PRÁCTICAS.....	35
3.4 EVALUACIÓN	43
3.4.1 EVALUACIÓN ECONÓMICA / EVALUACIÓN COSTO-BENEFICIO	43
3.4.2 MATRIZ DE BENEFICIOS E IMPACTO ORGANIZACIONAL	46
CAPÍTULO IV – REFLEXIÓN CRÍTICA DE LA EXPERIENCIA	48
CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES	50
5.1. CONCLUSIONES.....	50

5.2. RECOMENDACIONES	51
5.3 FUENTES DE INFORMACIÓN	53
GLOSARIO	55
SIGLARIO.....	57
ANEXOS.....	59

ÍNDICE DE FIGURAS

Figura 1. Organigrama de la entidad financiera	12
Figura 2. Árbol de problemas	17
Figura 3. Árbol de objetivos	19
Figura 4. Metodología Scrum con sus actores y artefactos	24
Figura 5. Imagen referencial de las reuniones con los usuarios de negocio.....	36
Figura 6. Diagrama de Arquitectura de solución.....	37

ÍNDICE DE TABLAS

Tabla 1	Experiencias profesional - Entidad Financiera	3
Tabla 2	Experiencia profesional - Consultora NTT DATA	5
Tabla 3	Experiencia profesional - Fractal Consultores TI.....	6
Tabla 4	Experiencias profesional – Comprobantes de Pago Electrónico S.A.C.	6
Tabla 5	Experiencia profesional - Gosucoders S.A.C.	7
Tabla 6	Experiencia profesional - Comparabien S.A.	7
Tabla 7	Formación académica profesional	8
Tabla 8	Cursos realizados	8
Tabla 9	Certificaciones	9
Tabla 10	Idiomas	9
Tabla 11	Otros conocimientos	9
Tabla 12	Actividades realizadas por cada perfil.....	15
Tabla 13	Tecnologías a utilizar en el proyecto	21
Tabla 14	Etapas del proyecto.....	24
Tabla 15	Roles del equipo para la ejecución del proyecto.	38
Tabla 16	HU (Historias de usuario) definidas por el Product Owner.....	39
Tabla 17	Tabla de perfiles informativos con su respectivo costo	43
Tabla 18	Matriz de beneficios sociales e impacto organizacional del proyecto.....	46

INTRODUCCIÓN

Hoy en día para las organizaciones el proceso de capacitación es muy importante, debido a que contribuye a mejorar la productividad y el rendimiento de los colaboradores al fortalecer sus habilidades y conocimientos. Muchas organizaciones y áreas dentro de las mismas organizaciones apuestan por dinamizar y transformar sus procesos apoyados en la tecnología.

En el sector financiero, la eficiencia operativa y la calidad del servicio dependen de manera directa de la rápida y correcta integración del talento humano. Por ello, la optimización de los procesos de inducción y capacitación constituye un activo estratégico para la continuidad y el crecimiento del negocio de la entidad financiera.

Para el caso de la entidad financiera carecía de una herramienta centralizada para el monitoreo del periodo de inducción de los asesores de servicio en las oficinas, en donde existía el riesgo de que la capacitación no fuera la misma en todas las geografías, esta falta de capacidad limitaba la velocidad de ingresos de nuevos, también existen otros procesos manuales que no eran muy eficientes.

El objetivo de este trabajo de suficiencia profesional es de mejorar el proceso de capacitación y poder brindar al equipo de formación una herramienta que les permita el seguimiento y control E2E (End-to-End o de extremo a extremo) del proceso de inducción, garantizando la estandarización de la capacitación en todas las geografías.

Para la solución implementada se realizó bajo el enfoque Agile (centrado en la entrega iterativa e incremental de software), utilizando la metodología Scrum (un marco de trabajo ágil para la gestión de proyectos complejos) que permitió establecer hitos importantes para la necesidad del usuario interno de la entidad financiera, esto apoyado con el soporte tecnológico de la nube como Google Cloud Platform (GCP) y los diferentes frameworks (marcos de trabajo) de vanguardia para la construcción de la aplicación web.

El presente trabajo de suficiencia profesional se ha estructurado en cinco capítulos principales:

En el CAPÍTULO I - TRAYECTORIA PROFESIONAL, En este capítulo se detalla la trayectoria profesional y la formación académica del autor.

En el CAPÍTULO II – CONTEXTO EN EL QUE SE DESARROLLA LA EXPERIENCIA, se describe la información de la empresa en donde el autor adquirió esta experiencia, incluyendo la misión, visión, organización, el cargo y las funciones desempeñadas.

En el CAPÍTULO III –ACTIVIDADES DESARROLLADAS, se presentará la definición del problema, los objetivos, el alcance, la metodología, los fundamentos teóricos, las etapas del proyecto y la implementación de la mejora realizada.

En el CAPÍTULO IV - REFLEXIÓN CRÍTICA DE LA EXPERIENCIA, se expone un análisis crítico de la experiencia del autor.

En el CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES, se detalla las conclusiones y recomendaciones del presente trabajo.

CAPÍTULO I – TRAYECTORIA PROFESIONAL

1.1 PRESENTACIÓN PROFESIONAL

El autor del presente trabajo tiene el grado de Bachiller en la carrera de Ingeniería de Software de la Universidad Nacional Mayor de San Marcos. Su experiencia con más de ocho años abarca desde el análisis, diseño, construcción, pruebas y mantenimiento del software, también ha desempeñado roles Front-End, Back-End, Líder Técnico y Scrum Master. Asimismo, durante su experiencia profesional el autor ha destacado por su perseverancia y dedicación para aprender diferentes habilidades en los roles que ha desempeñado.

A continuación, se desglosa la información referente a la experiencia profesional, el historial académico y las certificaciones obtenidas, comúnmente presentadas en formato de tabla para mayor claridad.

1.2 EXPERIENCIA PROFESIONAL

A continuación, en las siguientes tablas se detalla la experiencia del autor:

Tabla 1
Experiencias profesional - Entidad Financiera

Empresa – Periodo	Detalle de la experiencia
Entidad Financiera abril 2018 - Actualidad	Cargo: Software Developer Associate Funciones: Análisis y desarrollo de proyectos de software, siguiendo los estándares y buenas prácticas del ciclo de vida del software, usando diferentes tecnologías webs. Interactuar con herramientas de Inteligencia Artificial (IA) como Github Copilot para acelerar la productividad y el desarrollo de software.

Participar activamente en las ceremonias ágiles bajo la metodología Scrum para la planificación trimestral de proyectos.

Utilizar herramientas DevOps como Jira, Jenkins, SonarQube, Git, Bitbucket para la orquestación, el control de versiones y el despliegue eficiente de software.

Participación en el proyecto de formación de nuevos asesores, desempeñando el rol de Scrum Master, disponibilizando la solución en la nube (Google Cloud Platform).

Cargo: Software Developer Analyst

Funciones:

Construir servicios web en Java para el canal de la banca móvil personas.

Proveer soporte, atención de incidencias y mejoras de la aplicación banca móvil personas naturales utilizando las tecnologías como Android y iOS.

Coordinar y gestionar las atenciones de incidencias y mejoras con los proveedores de las fábricas digitales para asegurar la estabilidad del servicio de la banca móvil.

Participación en ceremonias ágiles, aplicando las metodologías Kanban y Scrum.

Controlar el versionado de código con Git, utilizando Bitbucket como repositorio.

Implementar mejoras en aplicaciones web de la zona publica de la banca por internet, empleando ReactJS y Spring MVC(Model-View-Controller).

Resolver incidencias productivas que impactan diferentes canales y áreas funcionales (Retail, Riesgos, Recursos humanos) en diferentes plataformas legacy.

Gestionar los despliegues a producción de las funcionalidades (mejoras, incidencias), asegurando que se cumpla con las

respectivas aprobaciones de los equipos de QA (Quality Assurance) y producción.

Fuente. Elaboración propia.

Tabla 2

Experiencia profesional - Consultora NTT DATA

Empresa – Periodo	Detalle de la experiencia
NTT DATA agosto 2017 – marzo 2018	Cargo: Solutions Assistant Cliente: Banco de Crédito del Perú Funciones: Desarrollar funcionalidades de la banca por internet, empleando tecnologías Front-End (AngularJS, TypeScript), Back-End (Spring Boot) y base de datos no relacional (MongoDB). Construir componentes web reutilizables, facilitando su uso y escalabilidad por otros equipos del proyecto. Participar activamente en las ceremonias ágiles de la metodología Scrum. Se integró un equipo Scrum como Team Member, desempeñando roles Front-End y Back-End durante los sprints del proyecto. Gestionar el control de versiones de código utilizando Git, y empleando GitLab como repositorio centralizado. Cargo: Solutions Assistant Cliente: Telefónica del Perú Funciones: Participación en un equipo Scrum como Team Member con el rol de Front-End. Desarrollar funcionalidades para la nueva aplicación del Contact Center con el Framework Angular 2.

Controlar las versiones de código con Git, utilizando Bitbucket como repositorio.

Fuente. Elaboración propia.

Tabla 3

Experiencia profesional - Fractal Consultores TI

Empresa – Periodo	Detalle de la experiencia
Fractal Consultores TI marzo 2017 – julio 2017	Cargo: Full Stack Software Developer Funciones: Desarrollar funcionalidades con tecnología Full Stack JavaScript (MeteorJS, AngularJS) y base de datos no relacional (MongoDB). Controlar las versiones de código con Git, aplicando la metodología GitFlow y utilizando Bitbucket como repositorio. Construir interfaces de usuario utilizando el Framework Bootstrap 3. Participar en las ceremonias ágiles e integrar un equipo Scrum como Team Member.

Fuente. Elaboración propia.

Tabla 4

Experiencias profesional – Comprobantes de Pago Electrónico S.A.C.

Empresa – Periodo	Detalle de la experiencia
Comprobantes de Pago Electrónico S.A.C. febrero 2016 – febrero 2017	Cargo: Full Stack Software Developer Funciones: Desarrollar componentes web y maquetar diseños utilizando HTML, Bootstrap (CSS). Consumir servicios API RESTFul empleando frameworks como AngularJS, Angular 2.

Implementar animaciones con JavaScript, y utilizar librerías de terceros.
Controlar las versiones de código mediante Git.
Construir servicios API RESTFul en lenguaje Golang.
Desplegar aplicaciones Front-End y Back-End en el producto AppEngine de Google Cloud Platform (GCP).

Fuente. Elaboración propia.

Tabla 5

Experiencia profesional - Gosucoders S.A.C.

Empresa – Periodo	Detalle de la experiencia
Gosucoders S.A.C. setiembre 2014 – mayo 2015	Cargo: Desarrollador de Aplicaciones Web Funciones: Construir las interfaces de usuario (UI) de los prototipos de aplicaciones, utilizando frameworks CSS como Bootstrap 3 y Materialize. Desarrollar APIs RESTFul en lenguaje PHP. Consumir servicios web (RestFul APIs) REST utilizando el Framework Backbone.js. Desplegar aplicaciones webs en servidores Apache.

Fuente. Elaboración propia.

Tabla 6

Experiencia profesional - Comparabien S.A.

Empresa – Periodo	Detalle de la experiencia
Comparabien S.A. febrero 2014 – agosto 2014	Cargo: Practicante Desarrollador Web Funciones: Generar informes para usuarios internos y ejecutivos, mediante la interfaz de phpMyAdmin sobre la base de datos en MySQL.

Diseñar y construir plantillas de correo electrónico para campañas digitales, utilizando HTML y CSS.
Desarrollar funcionalidades para un sistema web interno de recolección y gestión de información, utilizando el framework CodeIgniter y la base de datos en MySQL.

Fuente. Elaboración propia.

1.3 EDUCACIÓN Y CERTIFICACIONES

Tabla 7

Formación académica profesional

Formación	Institución	Periodo
Bachiller en Ingeniería de Software	Universidad Nacional Mayor de San Marcos	2010-2017

Fuente. Elaboración propia

Tabla 8

Cursos realizados

Institución	Curso
Executive Education CENTRUM PUCP	DevOps
Area 51 Training Center	FullStack JS
Educación Online CerTI+	Cloud Practitioner AWS
Repensar Educativo	UX Estratégico
Sistemas UNI	Administrador de Base de datos Oracle
DMC	Machine Learning Immersion

Fuente. Elaboración propia

Tabla 9
Certificaciones

Periodo	Certificación	Institución
2019	Scrum Master	Scrum Alliance

Fuente. Elaboración propia

Tabla 10
Idiomas

Idiomas	Institución de estudio	Nivel
Inglés	ICPNA	Intermedio

Fuente. Elaboración propia

Tabla 11
Otros conocimientos

Ofimática	MS Project, Microsoft Word, Excel, Point
Base de datos	SQL Server, Oracle, MYSQL, MongoDB
Lenguajes de programación	Java, Golang, PHP, Python, R, JavaScript
Plataforma Nube	Amazon Web Service, Google Cloud Platform

Fuente. Elaboración propia

CAPÍTULO II - CONTEXTO EN EL QUE SE DESARROLLA LA EXPERIENCIA

2.1 EMPRESA – ACTIVIDAD QUE REALIZA

La organización donde se desarrolló la presente experiencia corresponde a una sólida entidad financiera de reconocido prestigio tanto nacional como internacional.

Su propósito fundamental es brindar a sus clientes acceso a las oportunidades de esta nueva era. Para tal fin, la entidad apuesta por generar una experiencia que sea la más confiable, segura e innovadora. Esto la lleva a seguir impulsando la digitalización de sus procesos para mantenerse a la vanguardia del mercado financiero peruano y continuar ofreciendo productos que faciliten la vida de sus clientes.

La ejecución de sus principales funciones financieras se sustenta en una amplia cobertura territorial de oficinas en provincias y dependencias en el país.

2.2 VISIÓN

La entidad financiera busca constantemente un mejor futuro para las personas, un objetivo que se sostiene en tres valores claves:

- Integridad, se fundamenta en actuar con ética, honestidad y responsabilidad en todas las relaciones con sus grupos de interés (clientes, empleados y la sociedad), garantizando el estricto cumplimiento del marco legal.
- Prudencia, es obligatorio asegurar que la toma de decisiones en materia de riesgos sea cuidadosa y bien fundamentada, ya que esto es esencial para garantizar la solidez de la entidad financiera en el mercado.

- Transparencia, disponer de información clara, veraz y accesible para sus clientes y partes interesadas, esto es importante para obtener la confianza en la dirección de la entidad financiera.

2.3 MISIÓN

La misión de la entidad financiera es satisfacer las demandas de sus clientes, brindando los productos y servicios financieros de más alta calidad, un compromiso que mantiene desde hace años.

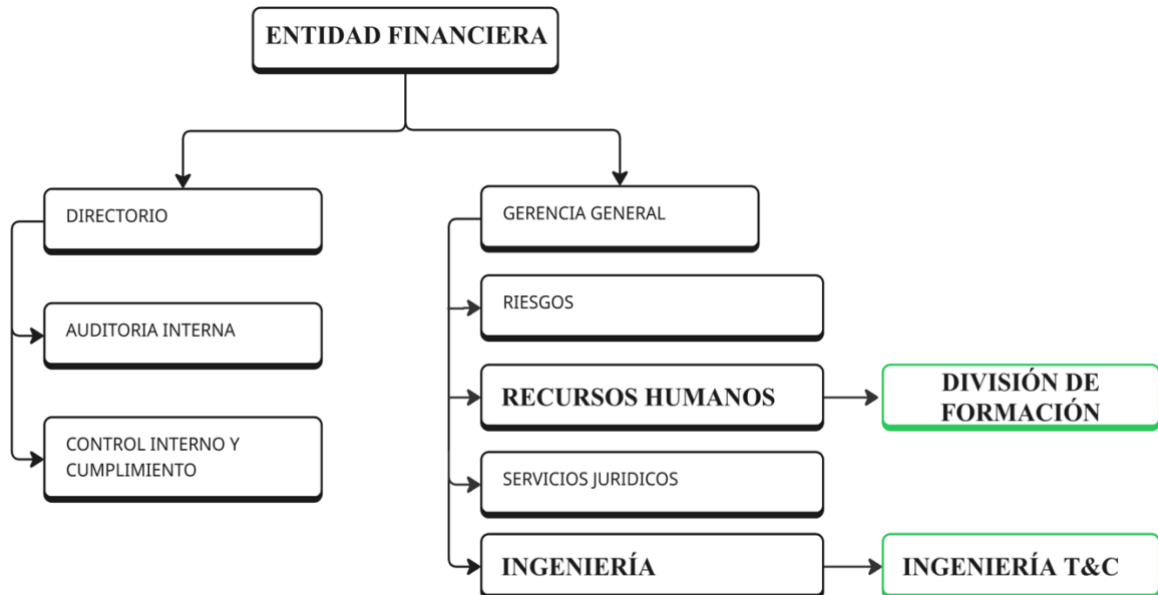
Por otro lado, busca aportar valor al capital de sus accionistas, sin descuidar el progreso y el bienestar de la población, teniendo como pilar fundamental mejorar la vida de las personas.

2.4 ORGANIZACIÓN DE LA EMPRESA

La entidad financiera se caracteriza por una estructura funcional con niveles jerárquicos definidos, como se visualiza en la Figura 1. No obstante, a nivel de gestión, la cultura corporativa está orientada a la horizontalidad operativa; esto conlleva que todas las unidades funcionales compartan un estatus de igual relevancia tras la Gerencia General, evitando silos informativos y promoviendo la equidad en la toma de decisiones estratégicas.

En la Figura 1, se presenta el organigrama de la entidad financiera. La presente experiencia profesional se lleva a cabo en coordinación con el área de recursos humanos para ayudar a su división de formación. El autor del presente trabajo se desempeñó en el área de ingeniería, en el equipo de ingeniería T&C (Talento y Cultura) por el cual se brindó todo el soporte tecnológico y técnico para la realización del proyecto.

Figura 1. Organigrama de la entidad financiera



Fuente. Elaboración propia

2.5 ÁREA, CARGO Y FUNCIONES DESEMPEÑADAS

En el marco de la presente experiencia profesional, el autor asumió las funciones de líder técnico dentro del área de Ingeniería de la entidad financiera, teniendo bajo su mando las siguientes tareas primordiales de gestión y desarrollo del proyecto, se describen a continuación:

- Coordinar con el área usuaria para poder aterrizar las necesidades o requerimientos del proyecto.
- Construir un proceso automatizado para la transferencia de información a los servidores de Google Cloud Platform (GCP).
- Gestionar el proyecto bajo el enfoque de la metodología Scrum, esto significó liderar las ceremonias ágiles para cumplir con los entregables del proyecto.

- Realizar las gestiones con otros equipos técnicos para los permisos y habilitación de infraestructura para los ambientes de test, calidad y producción.
- Asegurar el cumplimiento de las pruebas técnicas y funcionales con el equipo de Quality Assurance (QA) y el área usuaria respectivamente, esto con el fin de asegurar el compromiso y alcance del proyecto.
- Gestionar la instalación a los ambientes productivos de los componentes técnicos.
- Dar seguimiento y atención a las incidencias reportadas por el área usuaria.

2.6 EXPERIENCIA PROFESIONAL REALIZADA EN LA

ORGANIZACIÓN

El autor ha participado en diferentes proyectos y diferentes funciones que se detallan a continuación:

- Integró el equipo de desarrollo del proyecto de apertura de cuenta digital en la zona pública de la banca por Internet.
- Integró el equipo de desarrollo del proyecto de pago anticipado de préstamos en el canal de la banca por internet para personas y empresas.
- Participó en el desarrollo y pruebas del proyecto de migración de componentes y diseños para la nueva imagen de la entidad financiera en el canal de la banca por internet.
- Participó en el proyecto de capacitación y seguimiento de colaboradores para el área de formación como Scrum Master.
- Diseñar e implementar nuevas funcionalidades del canal retail web personas.
- Escribir test unitarios para garantizar la cobertura de código mayor al 80%.
- Utilizar herramientas de Inteligencia Artificial (GitHub Copilot, Gemini) para apoyarse en generar mayor productividad en la automatización de algunos procesos o fases del desarrollo de software.

- Construcción de casos de pruebas y ejecución siguiendo las técnicas y herramientas de pruebas de pares.
- Corrección de incidencias y estabilidad de las diferentes aplicaciones de los canales digitales como Agentes Corresponsales, Banca Móvil, Contact Center.
- Utilización de metodología ágiles para la gestión de proyectos e incidencias como Kanban y Scrum.
- Gestionar el proceso de pase a producción de las nuevas funcionalidades desarrolladas de acuerdo con los lineamientos de la organización.
- Elaborar la documentación respectiva de los desarrollos atendidos en cada trimestre.

CAPÍTULO III - ACTIVIDADES DESARROLLADAS

3.1 ANTECEDENTES

Dentro de la estructura del área de Recursos Humanos de la entidad financiera (EF), el equipo de formación profesional cumple la función estratégica de gestionar el desarrollo y capacitación de los nuevos asesores que serán destinados a la red de oficinas. Para este propósito, se implementa un proceso de inducción estructurado de forma secuencial con una duración aproximada de siete semanas. Durante este periodo, el equipo de formación designa un responsable o coordinador encargado de supervisar integralmente los aspectos técnicos y operativos previos a la inserción laboral definitiva de los nuevos asesores de servicio.

Cabe destacar que las vacantes de asesores de servicio presentan una alta rotación laboral debido a diversos factores, entre ellos: mejores ofertas laborales en el mercado, falta de motivación, incompatibilidad con las expectativas del puesto, entre otros. Esta situación genera que el proceso de inducción sea recurrente y continuo. A lo largo del proceso intervienen diferentes perfiles profesionales, estos se detallan en la Tabla 12.

Tabla 12
Actividades realizadas por cada perfil

Perfil	Actividades
Asesor de servicio	Es el colaborador que interactúa con los clientes en las oficinas y son los partícipes de los procesos de capacitación para garantizar una atención de calidad.
Equipo de formación	Responsable de gestionar los procesos de formación y poder generar cumplir con las capacitaciones de asesores de servicio.

Responsable de formación	Encargado de supervisar, gestionar materiales y recursos para que se puedan impartir en el proceso de formación.
Sub Gerente de Oficina	Supervisar operaciones, personal y la atención al cliente, asegurando que la oficina funcione de manera eficiente, también es responsable del nuevo asesor que ingresará a la oficina, se encarga de brindarle lo necesario para la atención y brindara su retroalimentación al nuevo asesor de servicio.

Fuente. Elaboración propia

Adicionalmente, el seguimiento de todo el proceso se encuentra centralizado en hojas de cálculo de Google, las cuales se actualizan de manera manual con información que vienen de diferentes fuentes como correos, encuestas, evaluaciones de otras plataformas.

3.2 SITUACIÓN PROBLEMÁTICA

En el sector financiero, la eficiencia operativa y la calidad del servicio dependen directamente de una integración rápida y efectiva del talento humano. Actualmente, el área de formación de la entidad financiera enfrenta desafíos críticos que limitan esta capacidad estratégica debido a la ausencia de una solución tecnológica centralizada para gestionar el ciclo de vida de capacitación del nuevo asesor de servicio.

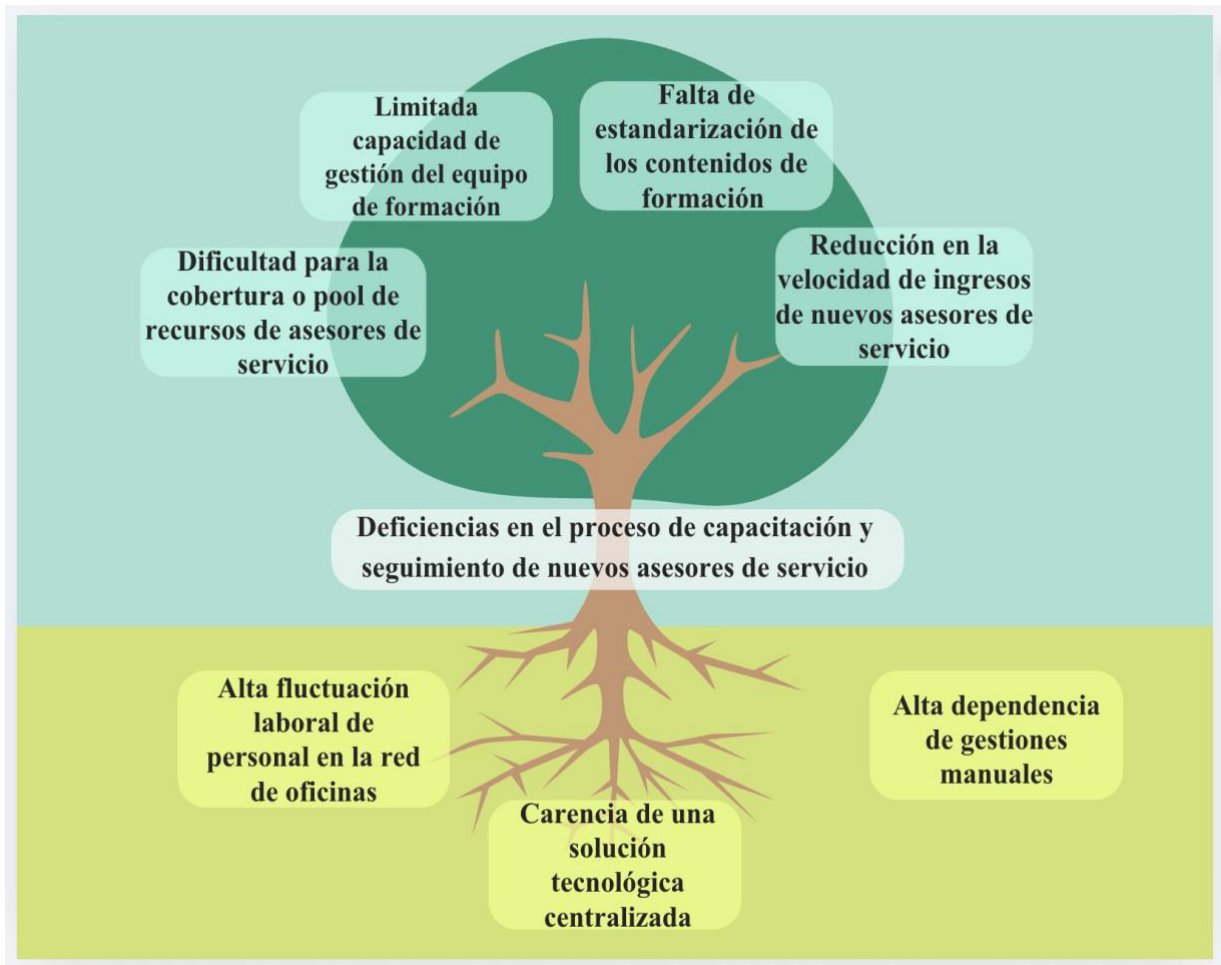
La problemática se manifiesta en cuatro dimensiones principales:

1. Fragmentación y Manualidad del Seguimiento: El proceso de inducción actual tiene una estructura secuencial con una duración aproximada de siete semanas, durante las cuales el seguimiento se realiza de forma manual mediante hojas de cálculo de Google (Google Sheets). La información que alimenta estas hojas proviene de fuentes heterogéneas como correos electrónicos, encuestas manuales y evaluaciones dispersas en otras plataformas, lo que genera una alta carga operativa para el equipo de formación y eleva el riesgo de errores en la consolidación de datos.

2. **Riesgo de Inconsistencia Geográfica:** Debido a la amplia cobertura territorial de la entidad, con oficinas en diversas provincias y dependencias a nivel nacional, existe el riesgo crítico de que la capacitación no sea homogénea. Al no contar con una plataforma centralizada que garantice el control E2E (extremo a extremo), la estandarización de los contenidos formativos se debilita, provocando que la calidad del aprendizaje dependa de la ubicación geográfica del asesor.
3. **Impacto de la Alta Rotación Laboral:** Las vacantes de asesores de servicio presentan una alta fluctuación de personal motivada por la competencia del mercado, falta de motivación o incompatibilidad con las expectativas del puesto. Esta recurrencia obliga a reiniciar el ciclo de capacitación constantemente; sin embargo, la dependencia de gestiones manuales limita la capacidad del equipo para cubrir las vacantes con la velocidad que el negocio requiere.

En resumen, la carencia de una solución tecnológica centralizada está generando ciertos procesos manuales afecten en la productividad del equipo de formación. En la Figura 2, se desarrolla el árbol de problemas, es una herramienta técnica empleada para identificar el problema central y secundarias. Este análisis nos ayudará a buscar una relación de tipo causa-efecto, permitiendo identificar con precisión los factores críticos que motivan a implementar una solución.

Figura 2. Árbol de problemas



Fuente. Elaboración propia

3.2.1 DEFINICIÓN DEL PROBLEMA

PROBLEMA PRINCIPAL

Deficiencias operativas y falta de trazabilidad en el proceso de capacitación y seguimiento de nuevos asesores de servicio.

PROBLEMAS ESPECÍFICOS

- Alta fluctuación laboral de personal en la red de oficinas: La recurrencia constante de los ingresos del personal, sumada a la lentitud de los procesos manuales, genera cuellos de botella que retrasan la disponibilidad de nuevos asesores en la red de oficinas.
- Carencia de una solución tecnológica centralizada: La ausencia de una solución única impide un monitoreo E2E (extremo a extremo), lo que imposibilita una visión integral del estado de la capacitación de los nuevos

asesores a nivel nacional.

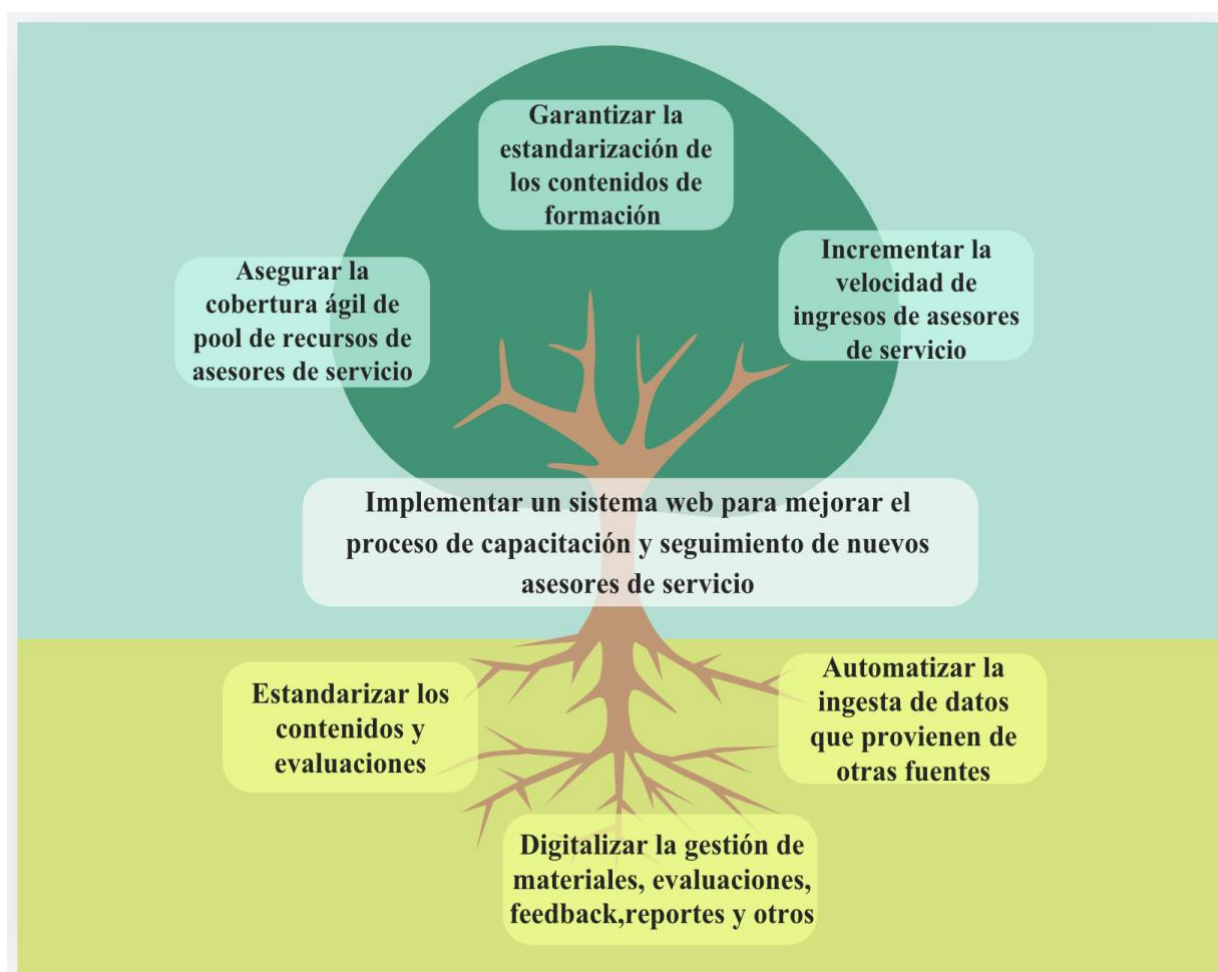
- Alta dependencia de procesos manuales: La dependencia de herramientas no integradas (Google Sheets y correos electrónicos) para gestionar un ciclo de siete semanas eleva la carga de trabajo y el riesgo de errores en la trazabilidad del progreso del nuevo asesor de servicio.

3.3 SOLUCIÓN

3.3.1 OBJETIVOS

Como se observa en la Figura 3, se realizó un árbol de objetivos con la finalidad de poder identificar los objetivos generales y específicos.

Figura 3. Árbol de objetivos



Fuente. Elaboración propia

A continuación, se identificó los siguientes objetivos:

3.3.1.1 OBJETIVO GENERAL

Implementar un sistema web para mejorar el proceso de capacitación y seguimiento de nuevos asesores de servicio, centralizando la gestión académica para garantizar la trazabilidad E2E (extremo a extremo).

3.3.1.2 OBJETIVOS ESPECÍFICOS

- Estandarizar los contenidos y evaluaciones: Unificar los materiales pedagógicos y las herramientas de medición del aprendizaje en una plataforma única. El fin es asegurar que todos los asesores, independientemente de su ubicación geográfica en la red de oficinas a nivel nacional, reciban una capacitación de idéntica calidad y bajo los mismos criterios de evaluación.
- Digitalizar la gestión de materiales, evaluaciones, feedback, reportes y otros: Migrar el flujo de trabajo actual (basado en documentos físicos o archivos dispersos) a un ecosistema digital interactivo. Esto busca mejorar la accesibilidad a los recursos de aprendizaje, asegurar la persistencia de los resultados históricos y facilitar el monitoreo en tiempo real por parte de los supervisores y formadores.
- Automatizar la ingesta de datos que provienen de otras fuentes: Implementar procesos automatizados (tareas programadas) para la transferencia y procesamiento de información desde otras plataformas, con el fin de eliminar la captura manual de datos en hojas de cálculo, reduciendo significativamente el error humano y la carga operativa del equipo de formación.

3.3.2 ALCANCE

3.3.2.1 ALCANCE FUNCIONAL

El alcance funcional del presente proyecto se ha dividido en 3 partes:

- **Gestión y estandarización de contenidos.**
 - El sistema permitirá la carga, administración y actualización centralizada de los materiales formativos (documentos, videos. Presentaciones, etc.).
 - La aplicación facilitará la asignación de los contenidos específicos según el rol y la ubicación geográfica del nuevo colaborador, esto depende del responsable a cargo de los grupos de formación.

- Se proveerá el acceso a la ruta de aprendizaje definida, el responsable a cargo de los grupos de formación velará que los contenidos sean los mismos para un mismo perfil o rol.

- **Monitoreo y control end-to-end(E2E)**

- La plataforma registrará y mantendrá actualizado el avance individual de cada colaborador en los módulos y materiales asignados.

- Se integrarán módulos de evaluación para medir el conocimiento adquirido, con calificación automática y registro histórico.

- Se proveerá al área de formación un dashboard para visualizar el estado actual del progreso de cada colaborador, permitiendo el control E2E.

- **Automatización de reportes y métricas**

- La aplicación automatizará la generación de reportes específicos de forma rápida, incluyendo métricas como seguimiento grupal o individual.

- El sistema almacenará los datos históricos de la capacitación para facilitar el análisis de desempeño y la toma de decisiones.

- Se incluirá la funcionalidad de notificación por correo electrónico para informar a los colaboradores sus estados de evaluaciones.

3.3.2.2 ALCANCE TÉCNICO

En la Tabla 13 se presenta un listado de las herramientas tecnológicas empleadas en el proyecto, destacando el uso de diversos servicios de Google Cloud Platform (GCP).

Tabla 13

Tecnologías a utilizar en el proyecto

Tecnologías	Detalle
Plataforma en la nube	El desarrollo de la solución se apoya en el soporte tecnológico de la nube, específicamente GCP.

Servicios en la nube	En el proyecto se han usado diferentes productos dentro del ecosistema de GCP como: Google App Engine, Google API Gateway, Cloud Storage, Cloud Sql, Cloud Build.
Lenguajes de programación y Frameworks	Para los componentes de la capa frontal se utilizó TypeScript con el framework VueJS. Para los componentes de la capa back se utilizó Java con el framework RestEasy.
Base de datos	Se optó por utilizar la base de datos MySQL
Herramientas DevOps	-Git -Bitbucket -Jira
Automatización de tareas	-Software Control-M
Componentes corporativos propios de la EF.	-API Envío de Correos -Tecnología Data Exchange -Tecnología SAML (Security Assertion Markup Language) -Plataforma de LMS (Learning Management System)

Fuente. Elaboración propia

3.3.2.3 ALCANCE ORGANIZACIONAL

El alcance organizacional de este proyecto se establece por iniciativa del área de formación de la entidad financiera, la cual es la unidad encargada de la gestión del talento humano y la calidad del servicio a través de la capacitación de los colaboradores que laboran en las oficinas.

3.3.2.4 ALCANCE GEOGRAFICO

El desarrollo del trabajo es a nivel del territorio nacional, al ser una aplicación de uso interno para los propios colaboradores de la entidad financiera.

3.3.3 ETAPAS Y METODOLOGÍA

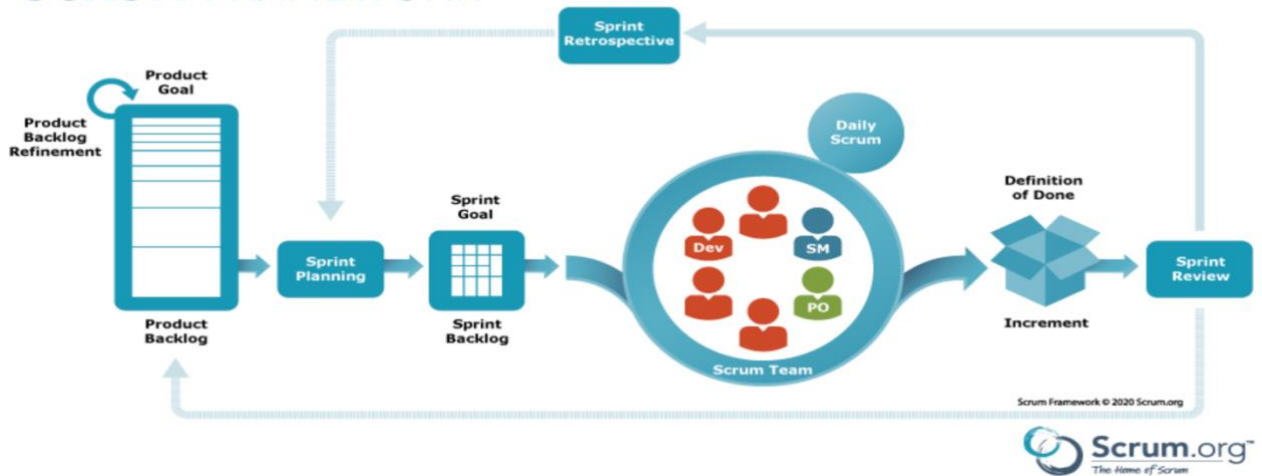
3.3.3.1 METODOLOGÍA

La ejecución del proyecto se gestionó bajo el marco de trabajo Scrum, el cual facilitó la entrega incremental de software funcional mediante ciclos denominados Sprints, con una duración establecida de 2 semanas cada uno. La gobernanza del proceso recayó en un equipo Scrum que se caracteriza por ser multidisciplinario en donde se definieron roles clave: el Product Owner, responsable de la priorización del Product Backlog; el Scrum Master, encargado de eliminar impedimentos; y el equipo de desarrollo, responsable de transformar los requerimientos en incrementos de valor. Durante cada iteración se celebraron las ceremonias de rigor: el Sprint Planning para definir los objetivos del ciclo, el Daily Scrum para el monitoreo diario de avances, y el Sprint Review, donde las funcionalidades fueron validadas por los usuarios finales y el Product Owner. Este enfoque iterativo permitió una adaptabilidad dinámica ante cambios en los requerimientos, asegurando que el producto final se alineara estrictamente con las necesidades operativas de la EF. Para el soporte administrativo y técnico, se adoptó una infraestructura de herramientas de nivel empresarial. La gestión y trazabilidad de las historias de usuario se centralizó en Jira, lo que permitió un monitoreo transparente de la velocidad del equipo y la salud de los Sprints. En paralelo, se utilizó Bitbucket como sistema de gestión de repositorios, facilitando el control de versiones, la revisión de código por pares (Code Review) y la colaboración técnica bajo estándares de integración continua.

En la Figura 4 se visualiza a forma marco la metodología.

Figura 4. Metodología Scrum con sus actores y artefactos

SCRUM FRAMEWORK



Nota: Fuente: Scrum Framework, tomado de la página web Scrum.org (2020).

3.3.3.2 ETAPAS

Los proyectos en la EF se rigen bajo un modelo de gestión por periodos trimestrales (Q1, Q2, Q3, Q4), esquema diseñado para la medición de hitos y avances en ciclos de corta duración. El proyecto se ejecutó en un lapso de un trimestre (1Q), articulándose en cinco fases principales como se observa en la Tabla 14.

Tabla 14
Etapas del proyecto.

Fases	Detalle
Fase Inicio	- Conformación el equipo Scrum que será el encargado de la ejecución del proyecto. -Reuniones con el área usuaria (equipo de formación) para poder entender sus necesidades o requerimientos. -Reuniones con el equipo de Arquitectura de soluciones para ver la viabilidad técnica de la solución planteada.

Fase Planificación	-Se definen las historias de usuario por parte del área de usuaria. -Priorización de las historias de usuario que serán parte de los sprints. -Estimaciones de las historias de usuario por parte del equipo. -Se definió la duración del proyecto en conjunto con la parte usuaria en aproximadamente un trimestre, esto acordando que cada sprint tiene una duración dos semanas, siendo un total de seis sprints para el desarrollo de todo el proyecto.
Fase Implementación	-Se constituyó el núcleo técnico del proyecto, donde se construyó el proceso automatizado en la nube y se gestionaron los permisos de infraestructura para los ambientes de test, calidad y producción. -El equipo de desarrollo trabaja en los entregables definidos en las historias de usuario. - Reuniones diarias para validar e identificar algún bloqueante que se tenga en el equipo. - Reuniones con el área usuaria al termino de cada sprint para validar los entregables. - Recolección data para la ejecución de las pruebas iniciales que se realizaran.
Fase Pruebas	-El equipo de QA ejecuta los casos de prueba de acuerdo las historias de usuario que se definieron. -El equipo de QA reporta las incidencias al equipo de desarrollo. -Reuniones con área usuaria usuario para reportar los estados de la fase de pruebas, además de poder tener feedback de lo que está validando. -El equipo de QA tiene que asegurar que no se tenga incidencias para

con ello dar su conformidad para la siguiente etapa.

- Fase Cierre**
- Gestionar el despliegue a producción de los componentes técnicos.
 - Coordinar con el área usuaria para la marcha blanca del proyecto.
 - Resolución de incidencias.

Fuente. Elaboración propia

3.3.4 FUNDAMENTOS UTILIZADOS

3.3.4.1. FUNDAMENTOS TEÓRICOS

Sistema Web

Los sistemas web se definen como aplicaciones de software que se ejecutan en servidores web y son accesibles mediante navegadores a través de internet o intranet. Según (Pressman & Maxim, 2015) , estas aplicaciones han evolucionado de simples páginas estáticas a sistemas complejos que integran bases de datos, lógica de negocio e interfaces interactivas.

Desde una perspectiva contemporánea, (Anu & Amadi, 2023) definen al sistema web como un entorno de información dinámico que trasciende la simple interfaz de usuario. Para estos autores, un sistema web es una configuración integral donde la tecnología, los procesos organizacionales y las personas convergen. Bajo este enfoque, el sistema no solo debe ser funcional a nivel de base de datos, sino que debe diseñarse bajo principios de aceptación tecnológica, donde la facilidad de uso y la utilidad percibida por el usuario final son los pilares que garantizan que la herramienta cumpla con los objetivos del negocio.

Se suele también llamar aplicación web a los sistemas web, se define como un software diseñado para ejecutarse a través de internet o una intranet mediante un navegador. Estas herramientas han cobrado gran relevancia debido a su agilidad y practicidad de uso desde cualquier explorador (Crea System, 2022).

MySQL

Se define como un sistema de gestión de base de datos relacionales que garantiza la persistencia de la información y permite el desarrollo de aplicaciones a gran escala gracias a su alta disponibilidad. Al respecto, (Google Cloud, 2026) señala que esta herramienta es fundamental en el ecosistema de aplicaciones modernas, destacando su capacidad para manejar transacciones críticas de manera rápida y confiable, lo que la convierte en una opción líder para entornos que requieren escalabilidad y seguridad de datos.

Según describe (Oracle, 2024), MySQL es un sistema de gestión de bases de datos relacionales (RDBMS) basado en SQL, además de ser la base de datos de código abierto más utilizado por las empresas líderes en tecnología, debido a su capacidad para ofrecer un alto rendimiento, fiabilidad y facilidades de uso.

Java

De acuerdo con (Mishra, Singh, & Agarwal, 2023) Java se consolida como un lenguaje de programación de propósito general, robusto y orientado a objetos, cuya arquitectura permite el desarrollo de aplicaciones que van desde sistemas embebidos hasta soluciones empresariales a gran escala, manteniendo siempre altos estándares de seguridad y escalabilidad.

Se define como una tecnología y plataforma informática que, desde su creación en 1995, ha evolucionado para convertirse en una base fiable para el desarrollo de servicios y aplicaciones en el entorno digital actual (Oracle, 2025)

Una de las mayores ventajas de Java es su independencia de la plataforma; gracias a la Máquina Virtual de Java (JVM), el código compilado puede ejecutarse en cualquier hardware o sistema operativo sin necesidad de ser modificado (IBM, 2026)

Servicios Web

Un servicio web es una aplicación de software identificada por un URI, cuyas interfaces se pueden definir y describir mediante XML. Este sistema soporta la interacción directa entre computadoras a través de una red, permitiendo que diferentes aplicaciones compartan datos e invoquen procesos independientemente de la plataforma en la que se encuentren (W3C, 2021).

Los servicios web constituyen una tecnología que permite el intercambio de información entre aplicaciones mediante el uso de protocolos abiertos. Según (Microsoft,

2024), se basan en estándares como HTTP para el transporte y formatos como XML o JSON para la estructuración de los datos, lo que garantiza que sistemas desarrollados en lenguajes distintos puedan comunicarse de manera efectiva.

Desde el punto de vista arquitectónico, un servicio web puede ser definido bajo el estilo REST (Representational State Transfer). De acuerdo con (IBM, 2024) estos servicios se caracterizan por ser ligeros, escalables y por utilizar los métodos estándar del protocolo HTTP (GET, POST, PUT, DELETE) para interactuar con los recursos de un sistema.

Arquitectura de servicios web

La arquitectura de servicios web se fundamenta en una estructura de capas que garantiza el intercambio de datos entre sistemas heterogéneos. Tal como señala (Red Hat, 2026) esta tecnología promueve la creación de servicios independientes y reutilizables. Dentro de las implementaciones actuales, se distinguen dos estilos principales: SOAP y REST. Mientras que SOAP ofrece un marco rígido de seguridad,

(Amazon Web Services, 2026) destaca que el modelo REST ha ganado predominancia gracias a su eficiencia en la transferencia de recursos mediante el protocolo HTTP, utilizando preferentemente formatos ligeros como JSON para el transporte de la información.

API (Application Programming Interface)

Las interfaces de programación de aplicaciones (API) actúan como puentes o intermediarios tecnológicos que facilitan la comunicación y el intercambio de datos entre programas de software distintos de manera predecible y documentada (Oracle, 2025)

Una API representa un mecanismo que permite la comunicación efectiva entre dos componentes de software. Según (Amazon Web Services, 2026) este intercambio se realiza mediante un conjunto de protocolos y definiciones que actúan como un contrato, donde una aplicación solicita información y la otra entrega una respuesta estructurada.

Google Cloud Platform

Se describe como un conjunto de servicios físicos y virtuales, tales como computadoras, unidades de disco duro y redes, que se encuentran distribuidos en los centros de datos de Google a nivel mundial. De acuerdo con (Google Cloud, 2024) esta

infraestructura permite a los desarrolladores desplegar aplicaciones en un entorno que ofrece redundancia, alta disponibilidad y la capacidad de escalar recursos de forma automática según la demanda del usuario.

Para aclarar el concepto de la redundancia en entornos de computación en la nube se refiere a la duplicación de componentes críticos del sistema (servidores, bases de datos y redes) para asegurar la continuidad del servicio ante fallos técnicos. Este mecanismo es el pilar de la alta disponibilidad, permitiendo que la aplicación permanezca operativa incluso cuando ocurren incidentes en la infraestructura.

Google App Engine

Es un producto de Google Cloud, ofrece un entorno de ejecución robusto que soporta múltiples lenguajes de programación, incluyendo Java, Python y Node.js. Según (Google Cloud, 2024), al utilizar entornos estándar o flexibles, los desarrolladores pueden desplegar microservicios de manera aislada y segura, garantizando que aplicaciones complejas mantengan un alto rendimiento y estabilidad en la nube.

Google API Gateway

Es un servicio totalmente gestionado que permite a los desarrolladores crear, publicar, mantener, supervisar y proteger APIs de forma segura a escala. Este funciona como un punto de entrada centralizado que gestiona el tráfico de las interfaces de programación de aplicaciones. Según (Google Cloud, 2026) este servicio permite exponer servicios backend de forma segura, encargándose de tareas críticas como la autenticación de usuarios, la limitación de velocidad y la gestión de cuotas, protegiendo así la integridad de los microservicios subyacentes.

Cloud Storage

Es un servicio gestionado para el almacenamiento de objetos de datos a escala mundial. Según (Google Cloud, 2026) este servicio permite almacenar y recuperar cualquier cantidad de datos en cualquier momento, ofreciendo una infraestructura que garantiza la integridad y disponibilidad de la información, siendo ideal para el alojamiento de archivos multimedia y copias de seguridad.

Cloud Sql

Se define como un servicio de base de datos totalmente gestionado que facilita la configuración, el mantenimiento y la administración de bases de datos relacionales en Google Cloud Platform. Según (Google Cloud, 2026) este servicio soporta motores como MySQL, PostgreSQL y SQL Server, ofreciendo características avanzadas de seguridad, alta disponibilidad y replicación automática para asegurar que los datos del sistema estén siempre protegidos y accesibles.

Cloud Build

Se define como una plataforma de integración y entrega continuas (CI/CD) que permite a los desarrolladores automatizar el proceso de construcción de aplicaciones en la nube. Según la documentación de (Google Cloud, 2026) este servicio ejecuta pasos de construcción de manera rápida y consistente en una infraestructura gestionada por Google, permitiendo que el código sea transformado automáticamente en artefactos listos para ser desplegados en servicios como App Engine.

Framework VueJS

Es un framework progresivo de JavaScript utilizado para la construcción de interfaces de usuario. A diferencia de otros frameworks integrales, está diseñado desde cero para ser adoptado de forma incremental, centrándose principalmente en la capa de la vista y facilitando la integración con otras librerías o proyectos existentes (Vue.js, 2026)

Framework RestEasy

Se define como una implementación de código abierto de la especificación JAX-RS (Java API for RESTful Web Services), diseñada para facilitar la creación de servicios web y aplicaciones REST en el entorno Java. Según (JBoss / Red Hat, 2025), este framework permite a los desarrolladores exponer lógica de negocio mediante protocolos HTTP, utilizando anotaciones simples de Java para gestionar peticiones y convertir datos automáticamente a formatos como JSON o XML.

Control-M

Es una plataforma de automatización de flujos de trabajo que permite a las organizaciones orquestar procesos de negocio complejos, integrando aplicaciones, datos y plataformas diversas. Según (BMC Software, 2026), esta herramienta proporciona una

interfaz centralizada para programar, monitorear y gestionar la ejecución de tareas, garantizando el cumplimiento de los acuerdos de nivel de servicio (SLA) mediante la detección proactiva de errores y la gestión de dependencias entre sistemas. Control-M simplifica la complejidad de la orquestación del flujo de trabajo, lo que facilita la definición, la programación, la gestión y la supervisión de los flujos de trabajo de aplicaciones complejas, lo que garantiza la visibilidad, la fiabilidad y la mejora de los SLA.

Git

Es un sistema de control de versiones distribuido que rastrea los cambios en el código fuente durante el desarrollo de software. De acuerdo con la documentación de Git (2026), su arquitectura permite que cada desarrollador posea una copia local completa del historial del proyecto, lo que facilita el trabajo sin conexión, la creación de ramas y la integración de cambios de manera no lineal y segura.

Bitbucket

Es una plataforma de gestión de repositorios basada en la nube que facilita la colaboración en equipo sobre el código fuente. Según (Atlassian, 2025) , más allá del simple almacenamiento, Bitbucket integra herramientas de revisión de código, gestión de permisos y canales de integración continua, permitiendo que los equipos de desarrollo centralicen sus activos digitales y aseguren la calidad del software antes de su despliegue final.

Jira Software

Es una herramienta de gestión de proyectos basada en la nube, diseñada para ayudar a los equipos a planificar, rastrear y lanzar productos de software de manera eficiente. Según (Atlassian, 2025), esta plataforma permite la visibilidad completa del ciclo de vida de desarrollo, facilitando la asignación de tareas, la gestión de errores (bugs) y el seguimiento del progreso mediante tableros ágiles, lo que asegura que el equipo cumpla con los objetivos del proyecto en los tiempos establecidos.

Tecnología SAML

De acuerdo con (IBM, 2025) , el estándar SAML consiste en un lenguaje basado en

XML desarrollado por OASIS que permite el intercambio de atributos de identidad y datos de seguridad. En un entorno operativo, esta tecnología facilita que un usuario se autentique ante un proveedor de identidades para obtener 'aserciones' o señales de seguridad. Estas aserciones son presentadas posteriormente a un proveedor de servicios para acceder a recursos de negocio, permitiendo establecer soluciones de inicio de sesión único (SSO) incluso cuando ambas entidades pertenecen a dominios de seguridad diferentes.

Plataforma LMS (Learning Management System)

Un Sistema de Gestión de Aprendizaje (LMS) se define como una plataforma tecnológica diseñada para centralizar la administración, distribución y gestión de programas de formación virtual. Según (ADR Formación, 2025), estas herramientas funcionan como un entorno organizado donde los administradores pueden gestionar inscripciones y materiales, mientras que los usuarios acceden a los recursos educativos de manera estructurada, permitiendo que la formación sea accesible y flexible según las necesidades de la organización.

Servicio de Programación de Tareas (Cron)

El servicio Cron en entornos de computación en la nube se define como un programador de tareas basado en el tiempo que permite la ejecución automática de procesos en intervalos específicos. De acuerdo con (Google Cloud, 2026) en la plataforma App Engine, este servicio funciona mediante el envío de solicitudes HTTP programadas a puntos de enlace (endpoints) definidos en la aplicación.

El uso de tareas programadas o Cron es fundamental para garantizar la autonomía operativa del sistema. Según la documentación de (Google Cloud, 2026), este servicio permite que el backend desarrollado en algún lenguaje de programación sea Java, Python, Golang, PHP, JavaScript del lado del servidor, se ejecute de forma automática la lógica de importación y procesamiento de datos en intervalos predefinidos.

Tecnología Data Exchange

El intercambio de datos se define como el sistema circulatorio de las organizaciones modernas, permitiendo el flujo de información entre sistemas, plataformas y partes interesadas para fundamentar la toma de decisiones. Según IBM (2025), en escenarios

donde no existen APIs modernas disponibles, el intercambio programado o por lotes (batch) mediante transferencias de archivos en la nube sigue siendo un método confiable y robusto. Este proceso permite unificar datos fragmentados, garantizando que la información crítica del negocio.

3.3.4.2. Fundamentos metodológicos

Scrum

Según la definición de (Schwaber & Sutherland, 2020) en la Guía de Scrum, este no es un proceso o una técnica estricta, sino un marco de trabajo dentro del cual se pueden emplear diversos procesos y técnicas. Su enfoque se centra en la mejora continua y la entrega de valor constante mediante el empirismo, lo que permite que el desarrollo del sistema de seguimiento de capacitación se adapte a los cambios en los requisitos del negocio.

De acuerdo con (Amazon Web Services, 2026), Scrum es un marco de gestión diseñado para que los equipos de trabajo se organicen de manera autónoma en función de metas compartidas. Se describe como un sistema que integra reuniones específicas, herramientas y roles determinados para garantizar una entrega de proyectos eficiente. La filosofía de Scrum, según esta perspectiva, se basa en la capacidad de los equipos para aprender de las experiencias previas, autogestionarse y responder con agilidad ante los cambios del entorno, permitiendo resolver problemas complejos de forma sostenible y rentable.

Pilares de Scrum

De acuerdo con (Schwaber & Sutherland, 2020), el marco de trabajo Scrum se fundamenta en el empirismo y se sustenta en tres pilares principales que permiten el control de procesos:

Transparencia: El proceso y el trabajo deben ser visibles tanto para quienes realizan el trabajo como para quienes lo reciben. Las decisiones importantes se basan en el estado percibido de sus tres artefactos formales; cuando la transparencia es incompleta, estas decisiones pueden ser erróneas y aumentar el riesgo.

Inspección: Los artefactos de Scrum y el progreso hacia los objetivos acordados deben inspeccionarse con frecuencia y diligencia para detectar variaciones o problemas potencialmente indeseables. La inspección permite la adaptación y se realiza de forma continua a través de los eventos del marco de trabajo.

Adaptación: Si durante la inspección se determina que uno o más aspectos de un proceso se desvían fuera de los límites aceptables, o que el producto resultante es inadecuado, el proceso o los materiales que se están procesando deben ajustarse. El ajuste debe realizarse lo antes posible para minimizar mayores desviaciones.

Responsabilidades en Scrum

Para Scrum, no existen diferencias, sino un equipo cohesionado, comprometido y enfocado en obtener un mismo objetivo. Según (Schwaber & Sutherland, 2020), el equipo Scrum consta de tres responsabilidades específicas:

Product Owner: Responsable de maximizar el valor del producto y del trabajo del equipo de desarrollo. Gestiona el Product Backlog y asegura que todos comprendan los elementos de trabajo.

Scrum Master: Facilitador que ayuda al equipo a comprender y aplicar Scrum, elimina impedimentos y protege al equipo de interrupciones externas.

Equipo de Desarrollo: Profesionales autoorganizados que realizan el trabajo de entregar un incremento de producto potencialmente liberable al final de cada Sprint.

Eventos de Scrum

Los eventos de Scrum son bloques de tiempo (time-boxed) diseñados específicamente para habilitar la transparencia y la inspección de los artefactos. Según (Schwaber & Sutherland, 2020), el evento principal es el Sprint, que actúa como un contenedor para todos los demás eventos:

Sprint: Período de tiempo fijo (usualmente 2-4 semanas) durante el cual se crea un incremento de producto utilizable. Cada Sprint tiene una meta específica de lo que se construirá.

Spring Planning (Planificación del Sprint): Reunión donde el equipo define qué se entregará en el próximo Sprint y cómo se logrará el trabajo.

Daily Scrum: Reunión diaria de 15 minutos donde el equipo sincroniza actividades y planifica las próximas 24 horas.

Sprint Review (Revisión del Sprint): Inspección del incremento y adaptación del Product Backlog si es necesario, realizada al final del Sprint con stakeholders.

Sprint Retrospective (Retrospectiva del Sprint): Es el último evento donde el equipo tiene la oportunidad de que se inspeccione a sí mismo y cree un plan de mejoras para el próximo Sprint.

Artefactos de Scrum

Los artefactos de Scrum representan el valor o el trabajo realizado y están diseñados para maximizar la transparencia de la información clave. Según (Schwaber & Sutherland, 2020) , cada artefacto contiene un compromiso que garantiza que se proporcione información que mida el progreso:

Product Backlog: Lista ordenada de todo lo que podría ser necesario en el producto, es la única fuente de requisitos. El Product Owner es responsable de su contenido, disponibilidad y orden.

Sprint Backlog: Conjunto de elementos del Product Backlog seleccionados para el Sprint, más un plan para entregar el incremento y cumplir el objetivo del Sprint.

Incremento: Suma de todos los elementos del Product Backlog completados durante un Sprint y todos los Sprints anteriores, debe estar en condición utilizable. Su compromiso es la Definición de Terminado (Definition of Done).

3.3.5 IMPLEMENTACIÓN DE LAS ÁREAS, PROCESOS, SISTEMAS Y BUENAS PRÁCTICAS

En esta sección se describen detalladamente las fases que conformaron la

ejecución de la solución, proceso que tuvo una duración estimada de tres meses. El autor mantuvo un involucramiento integral en cada etapa, desde la planificación inicial hasta el cierre del proyecto, ejerciendo el rol de Scrum Master para facilitar la gestión del equipo, liderar las ceremonias ágiles y asegurar el cumplimiento de los objetivos propuestos, además debido a la naturaleza técnica de su cargo como parte del área de Ingeniería brindó el soporte correspondiente al área usuaria de Recursos Humanos para que la solución cumpla con las necesidades funcionales que necesitaba dicha área.

3.3.5.1 Fase Inicio

En esta etapa preliminar, se establecieron las bases estratégicas y técnicas necesarias para garantizar la viabilidad del proyecto, asegurando la alineación entre las capacidades tecnológicas y las necesidades del negocio, estas se dividen en secciones para poder dar más contexto a la fase de inicio en las siguientes actividades :

Entendimientos de necesidades

Se llevaron a cabo de sesiones de trabajo con el área usuaria (equipo de formación). El objetivo fue identificar los puntos críticos o dolencias en el proceso actual del proceso de seguimiento y capacitación de los nuevos asesores de servicio, en la Figura 5 se muestra una imagen referencial de las reuniones que se tuvieron con los usuarios del área de formación.

Figura 5. Imagen referencial de las reuniones con los usuarios de negocio

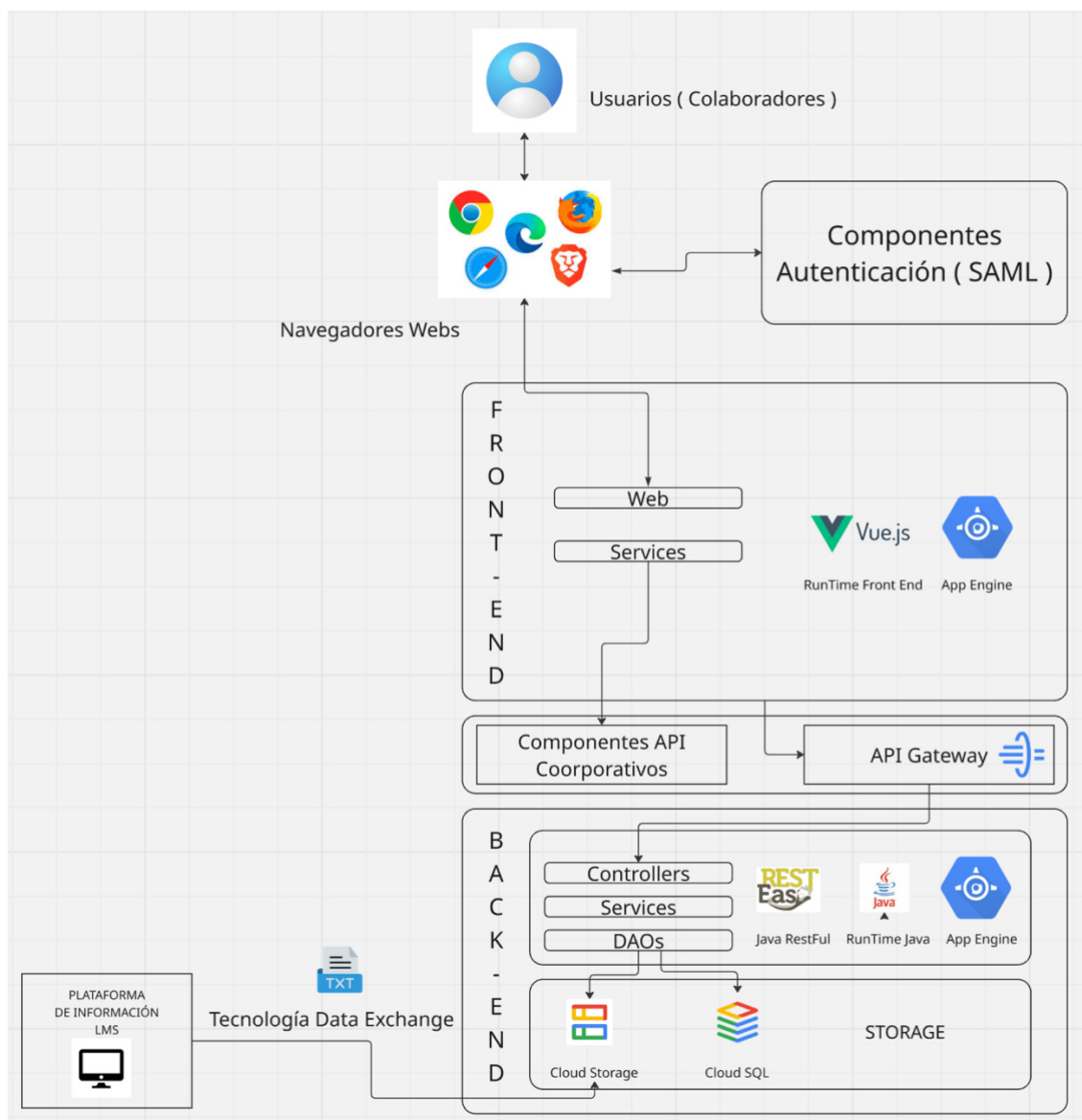


Nota : Fuente itockphoto.com (2026)

Análisis de Viabilidad Técnica

Se establecieron mesas técnicas con el equipo de Arquitectura de Soluciones. En este proceso, se validó la compatibilidad de la infraestructura de Google Cloud (App Engine y Cloud SQL) con los protocolos de seguridad de la organización. Se confirmó la viabilidad de implementar una arquitectura orientada a servicios en Java que pudiera orquestar las tareas mediante Cron Jobs y gestionar la autenticación federada vía SAML. El resultado que se obtuvo de las mesas de trabajo, inclusive con usuarios de negocio para poder sensibilizar lo necesario de tener una arquitectura del sistema que sea escalable y cubra con los estándares de la organización fue el diagrama de los componentes, en la Figura 6 se visualiza el diagrama de la arquitectura de la solución.

Figura 6. Diagrama de Arquitectura de solución



Nota: Fuente Elaboración propia

Conformación del Equipo Scrum

Se estructuró el equipo de trabajo definiendo las responsabilidades clave para la ejecución del proyecto. Se designó un Product Owner (representante del área de formación) para la priorización del valor, un Scrum Master para facilitar el proceso y asegurar la agilidad, y el equipo de Desarrollo (Developers) encargado de la construcción de los componentes técnicos. En la Tabla 15 se visualiza el equipo que participó en la ejecución del proyecto.

Tabla 15

Roles del equipo para la ejecución del proyecto.

Roles del equipo	Cantidad
Product Owner	1
Scrum Master	1
Desarrollador Front-End	1
Desarrollador Back-End	1
Desarrollador Data X	1
Especialista en Control-M	1
Analista de calidad	1

Fuente: Elaboración propia

3.3.5.2 Fase Planificación

En esta etapa, el objetivo principal fue traducir las necesidades del área de formación en un plan de ejecución técnico y temporal. En el rol de Scrum Master, se facilitó la creación de los artefactos de planificación y se aseguró que el equipo técnico y el área usuaria hablaran el mismo lenguaje.

Definición y Refinamiento del Product Backlog

A través de sesiones de trabajo, se procedió a la definición de las Historias de Usuario (HU). Estas historias fueron redactadas desde la perspectiva del área usuaria (equipo de

formación), las cuales se listan en la Tabla 16.

Tabla 16

HU (Historias de usuario) definidas por el Product Owner

CÓDIGO HU	Historia de Usuario
PAFYEDA-001	Extracción de datos desde la plataforma LMS.
PAFYEDA-002	Transferencia de información al servidor de la Entidad Financiera.
PAFYEDA-003	Ingesta y persistencia de datos en la base de datos.
PAFYEDA-004	Interfaz de bienvenida y control de acceso por rol.
PAFYEDA-005	Interfaz de gestión operativa de grupos , eventos y trazabilidad de avance académico.
PAFYEDA-006	Interfaz de dashboard de control informativo de grupos, usuarios , cursos.
PAFYEDA-007	Interfaz de consulta de progreso de los asesores de servicios.
PAFYEDA-008	Interfaz de reportes de los asesores de servicios.
PAFYEDA-009	Interfaz de gestión operativa de cursos y evaluaciones.
PAFYEDA-010	Interfaz de cursos y evaluaciones para los asesores de servicios.
PAFYEDA-011	Interfaz de entrenamiento en puesto de trabajo.

Fuente : Elaboración propia

Priorización por Valor de Negocio

Una vez definidos la lista de items del Product Backlog, se realizó la priorización del por parte del Product Owner, con ello se determinó qué funcionalidades debían desarrollarse primero para mitigar riesgos técnicos y entregar valor inmediato.

Estimación del Esfuerzo Técnico

El equipo de desarrollo realizó la estimación de las Historias de Usuario. Se utilizaron técnicas de estimación relativa (como Story Points) para evaluar la complejidad de tareas como el parseo de archivos TXT, la integración de seguridad SAML y la lógica de los Cron Jobs. Este proceso fue fundamental para asegurar que la carga de trabajo asignada a cada ciclo fuera realista.

Estructura Temporal y Cadencia del Proyecto (Sprints)

Se definió la hoja de ruta del proyecto bajo los siguientes parámetros temporales

acordados con los interesados:

- Duración total: un trimestre (aprox. tres meses).
- Duración del Sprint: dos semanas de trabajo constante.
- Cantidad de iteraciones: Un total de seis Sprints, lo que permitió establecer un ritmo de entrega predecible y puntos de control frecuentes para la inspección y adaptación del software.

3.3.5.3 Fase Implementación

En esta etapa se ejecutó la construcción física de la solución, integrando el desarrollo de software con la infraestructura en la nube. En el rol de Scrum Master, el enfoque principal consistió en mantener el ritmo de entrega (velocidad) y asegurar que los artefactos técnicos cumplieran con la "Definición de Terminado".

Configuración de Infraestructura y Entornos

Se constituyó el núcleo técnico mediante la gestión y aprovisionamiento de recursos en Google Cloud Platform (GCP). Se establecieron tres ambientes controlados para garantizar la estabilidad del ciclo de vida del software:

- **Desarrollo/Test:** Para las pruebas iniciales de código en Java.
- **Calidad (QA):** Donde el área de formación validó la funcionalidad de los reportes.
- **Producción:** Entorno final donde se desplegó el proceso automatizado mediante Cron Jobs. Este ordenamiento permitió que el equipo trabajara con permisos segregados, minimizando riesgos en la integridad de los datos.

Desarrollo Iterativo de Historias de Usuario

El equipo de desarrollo se centró en la construcción de los entregables priorizados. Se implementó la lógica de negocio en Java para el procesamiento de los archivos provenientes de DataX y se configuró la persistencia en Cloud SQL. Cada funcionalidad fue desarrollada siguiendo los criterios de aceptación definidos en la fase de planificación, asegurando que cada incremento de software fuera potencialmente entregable.

Gestión del Flujo de Trabajo (Eventos de Inspección)

El autor como responsable de la cumplir con la aplicación de la metodología, facilitó

los eventos clave para asegurar la transparencia:

- **Daily Scrums:** Se realizaron reuniones diarias de 15 minutos para sincronizar actividades y, fundamentalmente, identificar bloqueantes. El autor con el rol de Scrum Master, su prioridad fue gestionar la remoción de estos impedimentos (ej. retrasos en accesos a servidores o dudas en el formato de la data) para evitar desviaciones en el Sprint Backlog.
- **Sprint Reviews:** Al cierre de cada iteración, lideré las reuniones con el área usuaria. En estos espacios, el equipo demostró el software funcional, permitiendo que los interesados validaran los entregables y proporcionaran retroalimentación temprana, reduciendo el retrabajo.

Preparación para el Aseguramiento de Calidad

Se inició la recolección de data real (anonimizada) del LMS para la ejecución de las pruebas iniciales. Esta gestión fue crítica para validar que los algoritmos de Java procesaran correctamente los casos de borde y las excepciones en los registros de capacitación antes de pasar a la fase formal de testing.

3.3.5.4 Fase de Pruebas

En esta fase, el objetivo principal fue garantizar que el software fuera fiable, seguro y que cumpliera estrictamente con los criterios de aceptación. En el rol de Líder Técnico, la labor consistió en facilitar la comunicación entre el equipo de QA y Desarrollo para que las correcciones fluyeran sin retrasar el cronograma.

Ejecución de Casos de Prueba

El equipo de QA (Quality Assurance) procedió con la ejecución sistemática de los casos de prueba, diseñados en función de las historias de usuario establecidas en la fase de planificación. Se realizaron pruebas funcionales para validar que cada componente (carga en Java, persistencia en Cloud SQL y seguridad SAML) operara según lo esperado.

Gestión de Incidencias y Mejora Continua

Durante las iteraciones de prueba, el equipo de QA identificó y reportó las incidencias

(bugs) al equipo de desarrollo. Se facilitó un flujo de trabajo donde cada error fue categorizado por su severidad y prioridad, asegurando que los fallos críticos en la lógica de automatización o seguridad fueran resueltos de manera inmediata para no comprometer la integridad de la data de capacitación.

Comunicación con Stakeholders y Feedback

Se mantuvieron reuniones periódicas con el área usuaria para reportar el estado de la fase de pruebas. Estos espacios fueron fundamentales para obtener retroalimentación (*feedback*) sobre la usabilidad del sistema. Este contacto directo permitió que los usuarios finales validaran el comportamiento de los datos antes del cierre, asegurando que la solución fuera intuitiva y útil para sus procesos de formación.

Criterio de Aceptación y Conformidad

Como paso final previo al despliegue, el equipo de QA realizó un riguroso control para asegurar la ausencia de incidencias abiertas. La obtención de la conformidad de QA fue el hito que marcó la calidad del producto, garantizando que el sistema es robusto y está listo para ser promovido al ambiente de producción con total confianza.

3.3.5.5 Fase Cierre

Esta etapa final marcó la transición de la solución desde el entorno de desarrollo hacia el entorno operativo real de la organización. En calidad de Líder Técnico, la responsabilidad fue asegurar un despliegue sin interrupciones y facilitar la adopción tecnológica por parte del área usuaria.

Gestión del Despliegue a Producción

Se lideró la promoción de los componentes técnicos al entorno productivo de Google Cloud Platform. Esto incluyó el despliegue del backend en Java, la configuración final de las variables de entorno en App Engine y la activación de los Cron Jobs. Este proceso se realizó siguiendo protocolos de seguridad para garantizar que la transición no afectara otros servicios internos y que la conexión con el protocolo SAML fuera estable en el dominio corporativo.

Coordinación de la Marcha Blanca

Se estableció un periodo de marcha blanca en conjunto con el área usuaria (equipo de

formación). Durante este tiempo, el sistema operó en el entorno real pero bajo un monitoreo intensivo. Esta etapa fue crucial para validar que el procesamiento automático de los archivos de DataX se integrara perfectamente con los flujos de trabajo diarios de los usuarios, permitiéndoles familiarizarse con la nueva interfaz y las funcionalidades de automatización.

Estabilización y Resolución de Incidencias Post-Producción

Durante los primeros días de operación, se mantuvo un esquema de soporte para la resolución de incidencias críticas que pudieran surgir en el escenario productivo. El autor con el rol de Scrum Master, facilitó la comunicación directa entre el equipo de desarrollo y los usuarios para corregir ajustes menores de manera ágil. La fase cerró formalmente una vez que el sistema demostró estabilidad operativa y el área usuaria otorgó la conformidad final de la solución.

3.4 EVALUACIÓN

3.4.1 EVALUACIÓN ECONÓMICA / EVALUACIÓN COSTO-BENEFICIO

Evaluación Costo

El análisis económico de este informe se basa en valores referenciales detallados en la Tabla 17, dado el carácter reservado de los costos reales. Dicha simulación contempla el esfuerzo del equipo Scrum durante la vigencia del proyecto.

Tabla 17

Tabla de perfiles informativos con su respectivo costo

Perfiles	Unidades	Costo x Mes	Meses	Costo Total
Product Owner	1	S/ 12000	3	S/ 36000
Scrum Master	1	S/ 10000	3	S/ 30000
Desarrollador Front-End	1	S/ 7000	3	S/ 21000
Desarrollador Back-End	1	S/ 8000	3	S/ 24000
Desarrollador Data X	1	S/ 7000	3	S/ 21000

Especialista Control-M	1	S/ 7000	3	S/ 21000
Analista de Calidad	1	S/ 6000	3	S/ 18000
			Total	S/ 171000

Fuente: Elaboración propia

Como se puede visualizar, la inversión total del proyecto fue de S/ 171,000.00.

Beneficios Tangibles

Incremento en la Productividad y Eficiencia Operativa La capacitación continua del personal se traduce directamente en una reducción de tiempos de atención al cliente, disminución de errores operativos y optimización de procesos. Esto genera ahorros medibles en costos operativos y permite atender mayor volumen de transacciones con los mismos recursos.

Mejora en Indicadores Comerciales Un equipo capacitado logra mayores tasas de conversión en ventas de productos financieros, incremento en la cross-selling (venta cruzada) y up-selling. Los resultados se reflejan en el crecimiento de la cartera de clientes, aumento en la colocación de créditos y captación de depósitos.

Reducción de Costos por Rotación de Personal La inversión en capacitación disminuye significativamente los índices de rotación laboral, lo que representa ahorro en costos de reclutamiento, selección, inducción y el tiempo que toma alcanzar la productividad óptima de un nuevo colaborador.

Cumplimiento Normativo y Reducción de Riesgos La capacitación en temas regulatorios, prevención de lavado de activos, protección de datos y gestión de riesgos reduce potenciales multas, sanciones y contingencias legales que podrían afectar el patrimonio de la institución.

Retorno sobre la Inversión (ROI) Medible Los programas de capacitación permiten medir el retorno a través de indicadores como: aumento en ventas por empleado, reducción de quejas y reclamos, mejora en calificaciones de auditorías internas y externas, y disminución de pérdidas por fraudes.

Beneficios Intangibles

Posicionamiento y Prestigio Institucional Una entidad financiera reconocida por contar con profesionales altamente capacitados se diferencia en el mercado como una organización seria, confiable y orientada a la excelencia. Este posicionamiento atrae tanto a clientes como a talento humano de primer nivel, generando un círculo virtuoso de crecimiento.

Fidelización y Confianza del Cliente Los clientes perciben y valoran la calidad del servicio brindado por personal capacitado. La experiencia positiva en cada punto de contacto genera confianza, aumenta la satisfacción y promueve la lealtad a largo plazo. Un cliente bien atendido se convierte en embajador de la marca.

Motivación y Compromiso del Personal (Clientes Internos) La capacitación representa una inversión en el desarrollo profesional de los colaboradores, lo que eleva su autoestima, sentido de pertenencia y compromiso con la organización. Un empleado motivado es más productivo, proactivo y se convierte en promotor genuino de la cultura organizacional.

Cultura de Mejora Continua e Innovación Los programas de capacitación fomentan una mentalidad de aprendizaje permanente, apertura al cambio y búsqueda constante de mejores soluciones. Esta cultura de innovación permite a la entidad adaptarse rápidamente a las transformaciones del mercado financiero y tecnológico.

Clima Laboral Positivo La inversión en el desarrollo del talento humano mejora las relaciones interpersonales, fortalece el trabajo en equipo y crea un ambiente laboral favorable. Esto se refleja en menor ausentismo, mayor colaboración entre áreas y un ambiente propicio para la creatividad.

Reputación como Empleador Atractivo (Employer Branding) Ser reconocida como una institución que invierte en su gente atrae a los mejores profesionales del sector financiero. Esto facilita la atracción y retención del talento, posicionando a la entidad como empleador de elección en el mercado laboral.

Ventaja Competitiva Sostenible El conocimiento y las competencias desarrolladas en el equipo humano constituyen un activo estratégico difícil de replicar por la competencia.

Esta ventaja competitiva basada en el talento resulta más sostenible en el tiempo que las ventajas basadas únicamente en tecnología o productos.

3.4.2 MATRIZ DE BENEFICIOS E IMPACTO ORGANIZACIONAL

Debido a que el sistema es una herramienta de gestión interna, su evaluación no se basa en el retorno económico directo o utilidades por ventas. En su lugar, el análisis se centra en el valor estratégico y en la optimización de sus procesos de capacitación que la entidad brinda a sus nuevos asesores de servicios. Los beneficios de esta implementación se consideran de carácter indirecto, impactando positivamente en el bienestar del capital humano y en la eficiencia operativa, lo cual se traduce en una mejora sustancial de la experiencia del cliente final en la red de oficinas, en otras palabras, se sustenta en el ahorro operativo y en el impacto en los indicadores de satisfacción (NPS), los cuales a la fecha garantizan la sostenibilidad del proyecto a largo plazo.

Como se observa en la Tabla 18, los beneficios del sistema web trascienden el ámbito económico directo. El valor del proyecto reside en la optimización del capital humano y en la mejora de la calidad educativa. Al digitalizar el acompañamiento práctico y automatizar la trazabilidad de la información, se genera un impacto positivo en el clima laboral y, en última instancia, en la calidad del servicio ofrecido al cliente, cumpliendo así con los objetivos estratégicos de la entidad sin depender de un retorno de inversión monetario inmediato

Tabla 18

Matriz de beneficios sociales e impacto organizacional del proyecto

Categoría de Beneficio	Descripción del impacto	Resultado Esperado (Beneficio Indirecto)
Bienestar del colaborador	Reducción del estrés operativo mediante una plataforma intuitiva y centralizada.	Incremento en el compromiso y sentido de pertenencia del asesor.
Eficiencia en el Aprendizaje	Digitalización del proceso de formación permitiendo un avance autónomo y claro.	Reducción de la curva de aprendizaje y mayor rapidez para alcanzar la operatividad en ventanilla.
Calidad de la Mentoría	Estandarización del proceso de entrenamiento	Mejora en la relación responsable de formación-

	en oficina para que el sub gerente evalúe bajo criterios objetivos.	asesor y garantía de una formación homogénea a nivel nacional.
Experiencia del Cliente	Asesores mejor capacitados gracias a los contenidos actualizados.	Ejecución de una mejor experiencia en el servicio, mejorando los indicadores de satisfacción del cliente (NPS).
Trazabilidad y Transparencia	Disponibilidad de datos en tiempo real sobre el progreso de cada colaborador.	Eliminación de la subjetividad en las evaluaciones y procesos de promoción interna más justos.
Sostenibilidad Operativa	Sustitución de procesos manuales y uso de papel por un flujo digital en la nube.	Modernización tecnológica del área de formación y alineación con políticas de eco-eficiencia.

Fuente: Elaboración propia

CAPÍTULO IV – REFLEXIÓN CRÍTICA DE LA EXPERIENCIA

El desarrollo del presente proyecto representó para el autor un ejercicio de síntesis entre la formación académica y la práctica profesional, permitiendo validar que la ingeniería de software no solo consiste en la escritura de código, sino en la orquestación de tecnología, procesos y personas para generar valor de negocio.

A continuación, se presenta una reflexión analítica sobre los pilares que sustentan esta experiencia:

1. El empirismo de Scrum como catalizador de la horizontalidad Operativa: Aunque la organización mantiene niveles jerárquicos definidos, su cultura corporativa promueve la horizontalidad y la equidad en la toma de decisiones. En ese contexto, la implementación de Scrum no fue una ruptura con la estructura, sino una herramienta que potenció los pilares del empirismo (transparencia, inspección y adaptación). El autor pudo constatar que, antes del proyecto, la dependencia de gestiones manuales y “hojas de cálculo de Google” generaba precisamente esos “silos informativos” que la misión de la empresa busca evitar. Al introducir artefactos transparentes como el Product Backlog y el Sprint Backlog, se logró que el estatus del proyecto fuera visible para todas las unidades funcionales por igual, eliminando la subjetividad y permitiendo que la “horizontalidad operativa” se ejerciera sobre datos reales y no sobre procesos fragmentados.
2. Desafíos técnicos y escalabilidad en la nube: Desde una perspectiva técnica, la elección de una infraestructura en Google Cloud Platform(GCP) no fue una decisión trivial. El uso de servicios como App Engine y Cloud SQL permitió resolver el problema de la fragmentación geográfica. Sin embargo, la reflexión crítica reside en el proceso de aprendizaje continuo: a pesar de contar con bases sólidas universitarias, la integración de tecnologías emergentes y protocolos de seguridad corporativos, como el estándar SAML, representó un desafío técnico de alto nivel que obligó al autor a expandir su capacidad de autoaprendizaje y diseño de arquitecturas escalables.
3. Automatización como motor de eficiencia operativa: Un punto de inflexión en el proyecto fue la implementación de tareas programadas (Cron Jobs) y la orquestación mediante Control-M. La reflexión crítica permite identificar que la

verdadera mejora no radica solo en la digitalización, sino en la eliminación del error humano derivado de la captura manual de datos. Esta automatización liberó al equipo de formación de tareas repetitivas, permitiéndoles enfocarse en la calidad de la mentoría, lo que demuestra que la tecnología actúa como un habilitador estratégico para la sostenibilidad operativa de la entidad.

4. Evolución de habilidades blandas en entornos multidireccionales: Finalmente, el autor reconoce que el liderazgo técnico y la gestión del equipo Scrum potenciaron habilidades esenciales como la resolución de problemas complejos y la comunicación efectiva. La interacción con equipos multidisciplinarios (Ingeniería, Recursos Humanos y QA) subrayó que el éxito de un sistema de software depende de una comunicación fluida entre los stakeholders. Esta experiencia permitió al autor evolucionar de un perfil puramente técnico hacia uno de liderazgo integral, capaz de coordinar esfuerzos técnicos con objetivos estratégicos del sector financiero.

CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

La implementación del sistema web propuesto ha logrado cumplir satisfactoriamente con el objetivo general de mejorar el proceso de formación y seguimiento de nuevos asesores de servicio. A través de esta plataforma digital, se ha establecido un entorno integrado que optimiza la capacitación del personal, facilitando tanto la administración de contenidos como el monitoreo continuo del desempeño de los asesores en formación.

En cuanto a los objetivos específicos planteados, se han alcanzado los siguientes resultados:

Estandarización de contenidos y evaluaciones. El sistema ha permitido unificar los materiales de capacitación y las herramientas de evaluación, garantizando que todos los nuevos asesores reciban la misma calidad de formación independientemente del momento o lugar en que se incorporen. Esta estandarización ha eliminado las inconsistencias previas en el proceso formativo y ha establecido criterios claros y uniformes para medir el progreso y las competencias adquiridas.

Digitalización de la gestión integral. La transición de procesos manuales a digitales ha modernizado significativamente la administración de materiales, evaluaciones, retroalimentación y reportes. Esta digitalización no solo ha reducido el uso de documentos físicos y la carga administrativa, sino que también ha mejorado la accesibilidad, trazabilidad y organización de la información. Los supervisores y formadores ahora pueden gestionar y consultar datos de manera más eficiente, mientras que los asesores en formación tienen acceso inmediato a los recursos necesarios para su desarrollo profesional.

Automatización de la ingesta de datos. La integración del sistema con otras fuentes de información ha eliminado la necesidad de captura manual de datos, reduciendo errores y liberando tiempo valioso del personal. Esta automatización asegura que la información

relevante fluya de manera continua y confiable hacia el sistema, permitiendo una toma de decisiones más ágil y basada en datos actualizados.

En conjunto, estos logros representan un avance significativo en la profesionalización del proceso de capacitación de asesores de servicio, sentando las bases para una mejora continua en la calidad del servicio ofrecido a los clientes.

5.2. RECOMENDACIONES

1. Expansión de Funcionalidades con Inteligencia Artificial

Se recomienda implementar capacidades de inteligencia artificial en la plataforma para potenciar la experiencia de aprendizaje:

Chatbot inteligente: Desarrollar un asistente virtual que responda consultas frecuentes de los asesores en formación en tiempo real, reduciendo la carga del equipo de formación y proporcionando soporte 24/7.

Rutas de aprendizaje personalizadas: Utilizar algoritmos de machine learning para analizar el desempeño individual y sugerir contenidos específicos según las áreas de oportunidad de cada asesor.

2. Mejora Continua y Escalabilidad

Sistema de feedback 360°: Ampliar la funcionalidad de retroalimentación y encuestas para implementar un modelo de evaluación multidireccional. Esto permitirá que el asesor de servicio, el responsable de formación y el subgerente de oficina intercambien valores recíprocos sobre el desempeño y la calidad de los materiales, garantizando una visión integral que potencie la mejora continua y la adaptabilidad del proceso de capacitación.

3. Integración y Análisis de Datos

Dashboard ejecutivo avanzado: Crear visualizaciones interactivas con métricas clave (KPIs) que permitan a la alta gerencia tomar decisiones estratégicas basadas en datos de la capacitación.

Integración con otros sistemas corporativos: Conectar la plataforma con sistemas de gestión de talento humano, evaluación de desempeño y planificación de carrera para crear un ecosistema integrado.

Reportería avanzada con IA: Implementar capacidades de análisis automático que generen insights sobre tendencias, brechas de conocimiento y oportunidades de mejora en el programa formativo.

4. Gestión del Conocimiento

Base de conocimiento colaborativa: Crear un repositorio donde los asesores experimentados puedan compartir mejores prácticas, casos de éxito y lecciones aprendidas.

Comunidad de aprendizaje: Desarrollar foros temáticos y espacios de colaboración virtual que fomenten el aprendizaje entre pares y la construcción colectiva de conocimiento.

5. Medición de Impacto

Estudios de ROI periódicos: Establecer mecanismos para medir regularmente el retorno de inversión del sistema, vinculando métricas de capacitación con indicadores de desempeño en oficinas.

Encuestas de satisfacción: Implementar evaluaciones continuas de la experiencia del usuario (NPS) para identificar áreas de mejora en la plataforma.

5.3 FUENTES DE INFORMACIÓN

- Google Cloud. (s.f.). Obtenido de MySQL: <https://cloud.google.com/mysql>
- Oracle. (s.f.). Obtenido de <https://www.oracle.com/latam/mysql/what-is-mysql/>
- Barry, C. (2003). *Information Systems Development: Issues and Challenges*. New York: Kluwer Academic/Plenum Publishers.
- Pressman, R. S., & Maxim, B. R. (2015). *Ingeniería de software: Un enfoque práctico*. México D.F.: McGraw-Hill.
- ESICUniversity. (2025). *Aplicaciones web: qué son, tipos y ventajas*. Obtenido de <https://www.esic.edu/rethink/tecnologia/que-son-las-aplicaciones-web-c>
- Pressman, R. S., & Maxim, B. R. (2015). *Ingeniería de software: Un enfoque práctico*. México D.F.: McGraw-Hill.
- Anu, J., & Amadi, E. C. (2023). Development of a Design Theory for Web-Based Information Systems. *Journal of Research in Science*, 1(1), 12-25.
- Crea System. (22 de Septiembre de 2022). Obtenido de Crea System: <https://www.creasystem.net/posts/que-es-un-sistema-web>
- Google Cloud. (2026). Obtenido de Google Cloud: <https://cloud.google.com/mysql>
- Oracle. (2024). Obtenido de Oracle Latinoamerica: <https://www.oracle.com/latam/mysql/what-is-mysql/>
- Oracle. (2025). Obtenido de Java.com: https://www.java.com/es/download/help/whatis_java.html
- IBM. (2026). Obtenido de IBM Topics: <https://www.ibm.com/es-es/topics/java>
- Mishra, V., Singh, A., & Agarwal, S. (2023). Java's Preference: Exploring the Reason Behind Its Popularity. *ournal of Emerging Technologies and Innovative Research (JETIR)*, f495-f501.
- W3C. (2021). Obtenido de World Wide Web Consortium: <https://www.w3.org/TR/ws-arch/>
- Microsoft. (2024). Obtenido de Microsoft Learn: <https://learn.microsoft.com/en-us/troubleshoot/developer/visualstudio/cpp/general/web-services-overview>
- IBM. (2024). Obtenido de IBM Documentation: <https://www.ibm.com/docs/en/was-nd/9.0.5?topic=services-web-types-soap-rest>
- Red Hat. (2026). Obtenido de Red Hat: <https://www.redhat.com/es/topics/cloud-native-apps/what-is-service-oriented-architecture>
- Amazon Web Services. (2026). Obtenido de AWS: <https://aws.amazon.com/es/compare/the-difference-between-soap-rest/>
- Oracle. (2025). Obtenido de Oracle América Latina: <https://www.oracle.com/latam/cloud/cloud-native/api-management/what-is-api/>
- Amazon Web Services. (2026). Obtenido de AWS: <https://aws.amazon.com/what-is/api/>
- Google Cloud. (2024). Obtenido de Google Cloud Documentation: <https://docs.cloud.google.com/docs/overview?hl=es>
- Google Cloud. (2024). Obtenido de Google Cloud Documentation: <https://cloud.google.com/appengine/docs>
- Google Cloud. (2026). Obtenido de Google Cloud Documentation: <https://cloud.google.com/storage/docs?hl=es>
- Google Cloud. (2026). Obtenido de Google Cloud Documentation:

<https://cloud.google.com/api-gateway/docs?hl=es>

Google Cloud. (2026). Obtenido de Google Cloud Documentation:
<https://docs.cloud.google.com/sql/docs?hl=es>

Google Cloud. (2026). Obtenido de Google Cloud Documentation:
<https://cloud.google.com/build/docs?hl=es>

Vue.js. (2026). Obtenido de Vue.js - The Progressive JavaScript Framework:
<https://vuejs.org/guide/introduction.html>

JBoss / Red Hat. (2025). Obtenido de RESTEasy Documentation:
https://docs.resteasy.dev/6.2/userguide/#_overview

BMC Software. (2026). Obtenido de Control-M Documentation:
<https://www.bmc.com/it-solutions/control-m.html#overview>

Atlassian. (2025). Obtenido de Bitbucket Guides:
<https://www.atlassian.com/software/bitbucket/guides/getting-started/overview>

Atlassian. (2025). Obtenido de Jira Software Guides:
<https://www.atlassian.com/es/software/jira/guides/getting-started/introduction>

IBM. (2025). Obtenido de IBM Documentation: <https://www.ibm.com/docs/es/was-zos/8.5.5?topic=concepts-saml>

ADR Formación. (2025). *ADR*. Obtenido de Blog de ADR Formación:
https://www.adrformacion.com/blog/que_es_un_lms_definicion_caracteristicas_y_beneficios.html

Google Cloud. (2026). Obtenido de Google Cloud:
<https://docs.cloud.google.com/appengine/docs/legacy/standard/java/config/cron-yaml?hl=es-419>

Google Cloud. (2026). Obtenido de Google Cloud Documentation:
<https://docs.cloud.google.com/appengine/docs/legacy/standard/java/config/cron-yaml?hl=es-419>

Schwaber, K., & Sutherland, J. (Noviembre de 2020). Obtenido de Scrum Guides:
<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf>

Amazon Web Services. (2026). Obtenido de AWS - Centro de conceptos de computación en la nube: <https://aws.amazon.com/es/what-is/scrum/>

GLOSARIO

Agile: Enfoque de gestión de proyectos que promueve la entrega iterativa e incremental de software mediante ciclos cortos de desarrollo y adaptación continua a los cambios.

API Gateway: Servicio en la nube que funciona como punto de entrada centralizado para gestionar, monitorear y proteger las interfaces de programación de aplicaciones (APIs).

App Engine: Plataforma como servicio (PaaS) de Google Cloud que permite desarrollar y alojar aplicaciones web sin gestionar la infraestructura subyacente.

Arquitectura orientada a servicios: Modelo de diseño de software donde las funcionalidades se organizan como servicios independientes e interoperables que se comunican a través de protocolos estandarizados.

Artifact (Artefacto): En Scrum, representación tangible del trabajo o valor generado (Product Backlog, Sprint Backlog, Incremento).

Backend: Capa del sistema que contiene la lógica de negocio, gestión de bases de datos y procesamiento de información, no visible directamente para el usuario final.

Base de datos relacional: Sistema de almacenamiento de información organizada en tablas relacionadas mediante claves, que permite realizar consultas complejas mediante SQL.

Bitbucket: Plataforma de alojamiento de código basada en Git que facilita la colaboración, revisión de código e integración continua.

Bootstrap: Framework CSS que proporciona componentes prediseñados y un sistema de cuadrícula responsivo para facilitar el desarrollo de interfaces web.

Capacitación: Proceso sistemático y planificado de formación dirigido a desarrollar competencias específicas en los colaboradores de una organización.

Cloud Build: Servicio de Google Cloud que automatiza la construcción, prueba y despliegue de aplicaciones mediante pipelines de CI/CD.

Cloud SQL: Servicio de base de datos relacional completamente administrado en Google Cloud que soporta MySQL, PostgreSQL y SQL Server.

Cloud Storage: Servicio de almacenamiento de objetos en la nube de Google que permite guardar y recuperar datos de forma escalable y duradera.

Control-M: Plataforma empresarial de automatización y orquestación de flujos de trabajo que permite programar, monitorear y gestionar procesos de negocio complejos.

Cron Job: Tarea programada que se ejecuta automáticamente a intervalos de tiempo específicos en sistemas Unix/Linux o en plataformas cloud.

Daily Scrum: Reunión diaria de 15 minutos donde el equipo Scrum sincroniza actividades e identifica impedimentos.

Data Exchange (DataX): Tecnología o proceso que facilita el intercambio automatizado de información entre sistemas mediante transferencia de archivos o APIs.

Definition of Done (Definición de Terminado): Conjunto de criterios que debe cumplir un incremento de producto para ser considerado completo y listo para su entrega.

DevOps: Conjunto de prácticas que combinan desarrollo de software (Dev) y operaciones de TI (Ops) para acortar el ciclo de vida del desarrollo y mejorar la entrega continua.

Empirismo: Filosofía de Scrum basada en que el conocimiento proviene de la experiencia y en tomar decisiones fundamentadas en lo observado.

SIGLARIO

API – Application Programming Interface (Interfaz de Programación de Aplicaciones)

AWS – Amazon Web Services

CSS – Cascading Style Sheets (Hojas de Estilo en Cascada)

CI/CD – Continuous Integration/Continuous Delivery (Integración Continua/Entrega Continua)

E2E – End-to-End (De extremo a extremo)

EF – Entidad Financiera

GCP – Google Cloud Platform (Plataforma de Google Cloud)

HTML – HyperText Markup Language (Lenguaje de Marcado de Hipertexto)

HTTP – HyperText Transfer Protocol (Protocolo de Transferencia de Hipertexto)

HU – Historia de Usuario

IA – Inteligencia Artificial

JAX-RS – Java API for RESTful Web Services

JSON – JavaScript Object Notation (Notación de Objetos de JavaScript)

JVM – Java Virtual Machine (Máquina Virtual de Java)

LMS – Learning Management System (Sistema de Gestión de Aprendizaje)

MVC – Model-View-Controller (Modelo-Vista-Controlador)

QA – Quality Assurance (Aseguramiento de Calidad)

RDBMS – Relational Database Management System (Sistema de Gestión de Bases de Datos Relacionales)

REST – Representational State Transfer (Transferencia de Estado Representacional)

ROI – Return on Investment (Retorno sobre la Inversión)

SAML – Security Assertion Markup Language (Lenguaje de Marcado para Confirmaciones de Seguridad)

SLA – Service Level Agreement (Acuerdo de Nivel de Servicio)

SOAP – Simple Object Access Protocol (Protocolo Simple de Acceso a Objetos)

SQL – Structured Query Language (Lenguaje de Consulta Estructurado)

SSO – Single Sign-On (Inicio de Sesión Único)

TI – Tecnología de Información

TIR – Tasa Interna de Retorno

UI – User Interface (Interfaz de Usuario)

URI – Uniform Resource Identifier (Identificador Uniforme de Recursos)

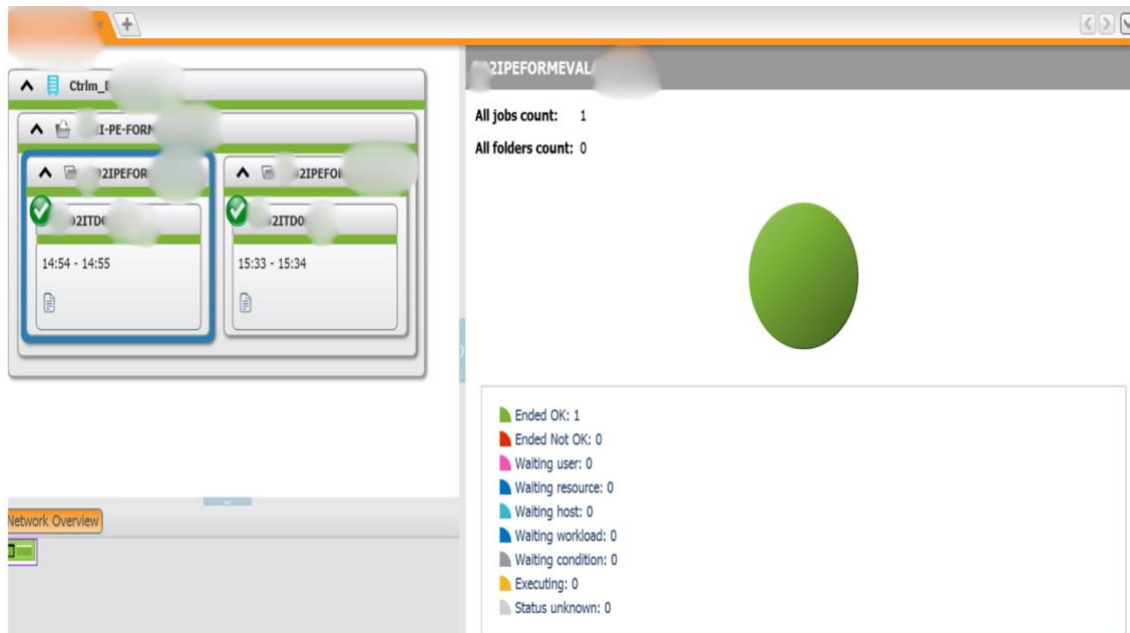
UX – User Experience (Experiencia de Usuario)

VAN – Valor Actual Neto

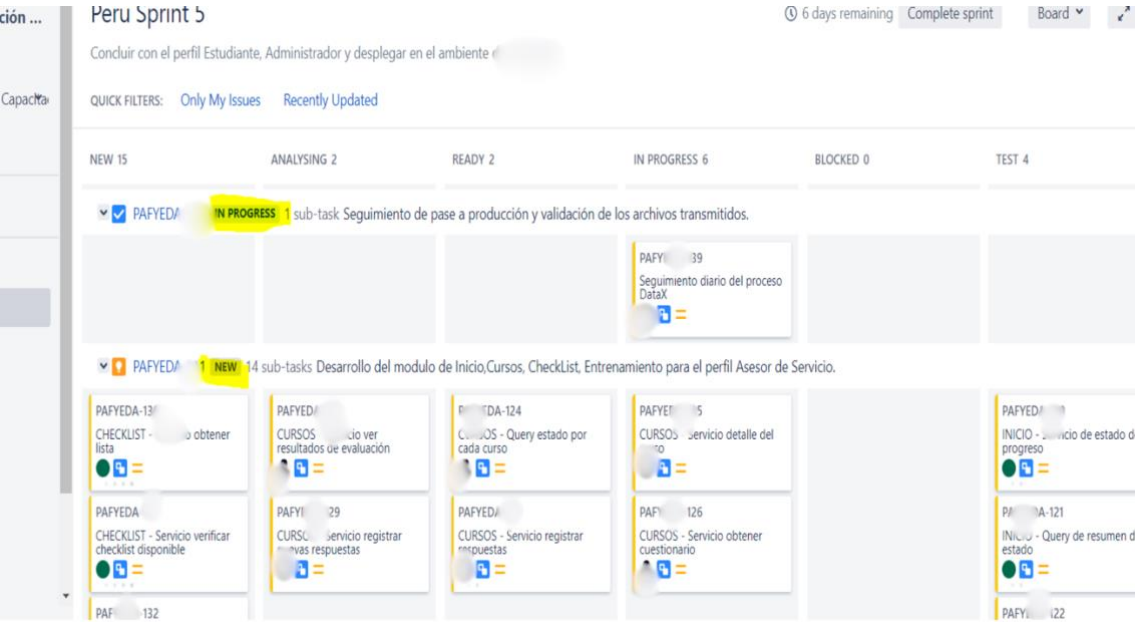
XML – eXtensible Markup Language (Lenguaje de Marcado Extensible)

ANEXOS

Anexo 1 : Entregable productivo del proceso de Control-M



Anexo 2: Tablero Jira en donde se llevó la gestión de las actividades del equipo



Anexo 3: Entregable productivo de transferencia de información de LMS a servidor de EF.

Google Cloud

-formevalasesore-

Buscar (/) recursos, documentos, productos y más

Buscar

Cloud Storage

Detalles del bucket

ACTUALIZAR

APRENDIZAJE

Buckets

Supervisión

Configuración

formevalasesore-bucket

Ubicación

us (varias regiones en Estados Unidos)

Clase de almacenamiento

Standard

Acceso público

No público

Protección

Ninguno

OBJETOS

CONFIGURACIÓNPERMISOSPROTECCIÓNCICLO DE VIDAOBSERVABILIDADINFORMES DE INVENTARIO

NUEVO

Depósitos

formevalasesore-bucket

SUBIR ARCHIVOS

SUBIR CARPETA
CREAR CARPETA
TRANSFERIR LOS DATOSADMINISTRAR CONSERVACIONESDESCARGARBORRAR

Filtrar solo por prefijo de nombre

FiltroFiltrar objetos y carpetas

Mostrar datos borrados

Nombre	Tamaño	Tipo	Fecha de creación	Clase de almacenamiento	Última modificación	Acceso público	Historial de versiones	Encriptación
Com..._Asignados.csv	106 KB	text/csv	12 15:55:04	Standard	12 15:55:04	No público	—	Administrada por Google
Com..._Finalizados.csv	572.9 KB	text/csv	12 16:33:55	Standard	12 6:33:55	No público	—	Administrada por Google
	85 B	text/csv	28 17:00:47	Standard	28 7:00:47	No público	—	Administrada por Google

Anexo 4: Casos de prueba que se ejecutaron en el proceso

Test Repository for project Peru Ap

Switch to Flat View

All (37)

Test Repository (37)

Seguimiento 0 (15)

Gestion de Usuarios 0 (12)

Gestion de Contenidos 0 (6)

Mailings 0 (3)

Vulnerabilidades 0 (1)

All

Filter

Showing 37 of 37 entries

ACCEPTED

PAFYEDA

eliminarGrupo - OK - Grupo eliminado y lista actualizada

MANUAL

ACCEPTED

PAFYEDA

agregarGrupo - OK - Muestra pantalla de alta y valida campos obligatorios

MANUAL

ACCEPTED

PAFYEDA

filtrarGruposPorNombre - OK - Filtra y muestra grupos por nombre

MANUAL

ACCEPTED

PAFYEDA

listarGrupos - OK - Retorna datos válidos

MANUAL

ACCEPTED

PAFYEDA

eliminarUsuario - OK - Usuario eliminado correctamente (individual o masivo)

MANUAL

ACCEPTED

PAFYEDA

agregarUsuario - OK - Usuario agregado correctamente (individual o masivo)

MANUAL

ACCEPTED

PAFYEDA

filtrarUsuariosPorNombreORegistro - OK - Filtra y muestra usuarios por nombre/registro

MANUAL