



# POSGRADOS

MAESTRÍA EN

SOFTWARE CON MENCIÓN EN  
DESARROLLO WEB Y MÓVIL

RPC-SO-34-NO.778-2021

OPCIÓN DE TITULACIÓN:

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

TEMA:

DESARROLLO DE UNA  
APLICACIÓN WEB QUE  
GESTIONE LAS ACTIVIDADES Y  
RECURSOS ASIGNADOS EN EL  
PLAN OPERATIVO ANUAL DEL  
INSTITUTO SUPERIOR  
UNIVERSITARIO ALMIRANTE  
ILLINGWORTH

AUTOR(ES)

ERNESTO ALONSO ROBLES PEÑAHERRERA

DIRECTOR:

MONICA DANIELA GOMEZ RIOS

GUAYAQUIL – ECUADOR

2025

**Autor(es):**



Ernesto Alonso Robles Peñaherrera  
Ingeniero en Sistemas Administrativo Computacionales  
Candidato a Magíster en Software por la Universidad Politécnica  
Salesiana – Sede Guayaquil.  
eralro@yahoo.es

**Dirigido por:**



Mónica Daniela Gómez Ríos  
Ingeniero en Sistemas  
Magister en Ciencias de la Computación Mención Networking  
Magister en Ciencias de la Computación Mención Aplicaciones  
Distribuidas  
mgomezr@ups.edu.ec

Todos los derechos reservados.

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

**DERECHOS RESERVADOS**

2025 © Universidad Politécnica Salesiana.

GUAYAQUIL– ECUADOR – SUDAMÉRICA

*Ernesto Alonso Robles Peñaherrera*

*Desarrollo de una Aplicación Web que Gestione las Actividades y Recursos Asignados  
en el Plan Operativo Anual del Instituto Superior Universitario Almirante Illingworth*

## **DEDICATORIA**

A mi esposa, compañera de vida y confidente, agradezco por ser mi faro en los momentos de oscuridad y mi compañía en los días de desafío. Tu paciencia, comprensión y aliento constante han sido esenciales en este viaje, y este logro no sería posible sin tu presencia constante.

A mis hijos por darme la motivación y ser ejemplo para ellos.

Finalmente, agradezco a Dios por la fortaleza que me ha otorgado para superar los desafíos y por guiarme a lo largo de este viaje educativo.

**Ernesto Alonso Robles Peñaherrera**

## **AGRADECIMIENTO**

Quisiera expresar mi sincero agradecimiento a todos aquellos que han contribuido de manera significativa a la realización de esta tesis de maestría.

A mis padres, quienes, aunque no puedan estar físicamente presente, siguen siendo una influencia fundamental, sus enseñanzas han sido mi mayor motivación, y perduran en mi memoria y en este trabajo.

A mi esposa e hijos, la paciencia, comprensión y aliento constante han sido fundamentales en cada etapa de este proyecto. Cada éxito alcanzado es compartido contigo, y valoro profundamente tu presencia en este viaje.

Además, agradezco a mis respetados profesores y mentores por su orientación experta y su dedicación inquebrantable. Sus valiosos consejos y enseñanzas han sido la brújula que ha guiado mi investigación hacia el éxito académico.

Este logro no es solo mío, sino de todos aquellos que han formado parte de este viaje. Gracias por su apoyo incondicional y por ser parte integral de esta etapa significativa de mi vida académica.

# TABLA DE CONTENIDO

Resumen .....	10
Abstract .....	11
1. Introducción .....	12
1.1 Antecedentes .....	12
1.2 Determinación del problema .....	15
1.2.1 Descripción del problema .....	15
1.2.2 Formulación del problema .....	16
1.2.3 Justificación del problema.....	17
1.2.4 Delimitación del problema.....	19
1.3 Justificación del problema .....	20
1.4 Objetivos .....	21
1.4.1 Objetivo General .....	21
1.4.2 Objetivo Específicos .....	21
2. Marco teórico referencial.....	22
2.1 Plan Operativo Anual .....	22
2.2 El Uso de Microservicios en el Desarrollo de Aplicaciones Web.....	23
2.2.1 Arquitectura monolítica .....	24
2.2.2 Microservicio .....	25
2.2.3 Desafíos y Consideraciones en la Implementación de Microservicios .....	27
2.3 Metodologías Ágiles en el desarrollo de software .....	28
2.3.1 Manifiesto Ágil .....	29
2.3.2 SCRUM.....	30
2.4 Integración Continua y Despliegue Continuo (CI/CD) .....	32
2.5 Definición y concepto de CI/CD .....	32
2.6 Importancia de CI/CD en el Desarrollo de Aplicaciones Web .....	34
2.7 Herramientas y Tecnologías .....	34
3. Materiales y metodología.....	36
3.1 Enfoque metodológico.....	36
3.2 Herramientas utilizadas .....	39
3.3 Levantamiento de requerimientos .....	43
3.4 Diseño del sistema .....	47

3.5	Implementación por sprints.....	50
3.6	Sprint 1 – Preparación Técnica e Infraestructura .....	51
3.7	Sprint 2 – Autenticación y Gestión de Usuarios .....	51
3.8	Sprint 3 – Gestión de Objetivos y Seguimiento de Metas .....	52
3.9	Sprint 4 – Desarrollo del Front-End .....	53
3.10	Sprint 5 – Medios de verificación y validaciones .....	54
3.11	Sprint 6 – generación de informes automáticos.....	54
3.12	Sprint 7 – CI/CD y automatización .....	55
3.13	Sprint 8 – documentación y pruebas finales.....	56
3.14	Validación funcional .....	56
4.	Resultados y discusión.....	58
4.1	Descripción general de la solución desarrollada .....	58
4.2	Módulos y funcionalidades implementadas .....	60
4.3	Validación funcional del sistema .....	63
4.4	Evaluación de cumplimiento de objetivos .....	68
5.	Conclusiones.....	70
	Referencias .....	71

# ÍNDICE DE TABLAS

---

Tabla 1. ....	17
Tabla 2. ....	26
Tabla 3. ....	31
Tabla 4. ....	37
Tabla 5. ....	37
Tabla 6. ....	40
Tabla 7. ....	43
Tabla 8. ....	44
Tabla 9. ....	47
Tabla 10. ....	50
Tabla 11. ....	51
Tabla 12. ....	52
Tabla 13. ....	52
Tabla 14. ....	53
Tabla 15. ....	54
Tabla 16. ....	55
Tabla 17. ....	55
Tabla 18. ....	56
Tabla 19. ....	57
Tabla 20. ....	60
Tabla 21. ....	66
Tabla 22. ....	68

# ÍNDICE DE ILUSTRACIONES

---

Ilustración 1.....	30
Ilustración 2.....	32
Ilustración 3.....	39
Ilustración 4.....	48
Ilustración 5.....	59
Ilustración 6.....	61
Ilustración 7.....	62
Ilustración 8.....	62
Ilustración 9.....	63
Ilustración 10.....	63
Ilustración 11.....	64
Ilustración 12.....	64
Ilustración 13.....	65

DESARROLLO DE UNA  
APLICACIÓN WEB QUE  
GESTIONE LAS  
ACTIVIDADES Y  
RECURSOS ASIGNADOS  
EN EL PLAN  
OPERATIVO ANUAL DEL  
INSTITUTO SUPERIOR  
UNIVERSITARIO  
ALMIRANTE  
ILLINGWORTH

AUTOR(ES):

ERNESTO ALONSO ROBLES  
PEÑAHERRERA

## RESUMEN

---

Este proyecto aborda la problemática de la gestión y seguimiento ineficiente del Plan Operativo Anual (POA) en el Instituto Superior Universitario Almirante Illingworth (AITEC), siendo el POA un instrumento estratégico derivado del Plan Estratégico de Desarrollo Institucional (PEDI), utilizado para alinear las acciones operativas con los objetivos institucionales y cumplir con las exigencias normativas del Consejo de Aseguramiento de la Calidad de la Educación Superior (CACES).

Actualmente, el proceso de gestión del POA se realiza de manera manual mediante archivos compartidos en Google Drive, lo cual genera retrasos, inconsistencias en la información, dificultades en la rendición de cuentas y riesgos de desalineación estratégica. Para contrarrestar estos desafíos, se propone el desarrollo de una aplicación web integral que centralice y optimice el proceso del POA mediante una arquitectura de microservicios, autenticación segura con cuentas institucionales y la adopción de metodologías ágiles, específicamente Scrum; por tanto, la aplicación permitirá la planificación, registro, seguimiento en tiempo real, carga de medios de verificación y visualización del progreso a través de un panel de control.

Los objetivos específicos del proyecto incluyen estimar requisitos, implementar prototipos iterativos y adoptar prácticas ágiles para garantizar flexibilidad y alineación con las necesidades institucionales.

La solución busca mejorar la eficiencia operativa, la transparencia, la rendición de cuentas y asegurar el cumplimiento de los estándares de calidad exigidos por los organismos reguladores., además, la implementación de CI/CD (Integración Continua y Despliegue Continuo) complementa el enfoque ágil para optimizar el desarrollo y despliegue del software seguro.

**Palabras clave:** Microservicios, Plan Operativo Anual, Aplicación web, Integración y Despliegue Continuo.

## ABSTRACT

---

This project addresses the problem of inefficient management and monitoring of the Annual Operational Plan (POA) at the Instituto Superior Universitario Almirante Illingworth (AITEC). The POA is a strategic instrument derived from the Strategic Plan for Institutional Development (PEDI), used to align operational actions with institutional objectives and comply with the regulatory requirements of the Council for Quality Assurance in Higher Education (CACES).

Currently, the POA management process is done manually through files shared on Google Drive, which generates delays, inconsistencies in information, difficulties in accountability and risks of strategic misalignment. To counteract these challenges, we propose the development of a comprehensive web application that centralizes and optimizes the POA process through a microservices architecture, secure authentication with institutional accounts and the adoption of agile methodologies, specifically Scrum; therefore, the application will allow planning, recording, real-time tracking, uploading of verification means and visualization of progress through a control panel.

The specific objectives of the project include estimating requirements, implementing iterative prototypes and adopting agile practices to ensure flexibility and alignment with institutional needs.

The solution seeks to improve operational efficiency, transparency, accountability and ensure compliance with the quality standards required by regulatory bodies. In addition, the implementation of CI/CD (Continuous Integration and Continuous Deployment) complements the agile approach to optimize the development and deployment of secure software.

Keywords: Microservices, Annual Operational Plan, Web Application, Integration and Continuous Deployment.

# 1. INTRODUCCIÓN

---

## 1.1 ANTECEDENTES

El Tecnológico Superior Almirante Illingworth, tiene un plan estratégico; el cual, “es un proceso vinculado a la toma de decisiones institucionales que debe ser continuo, reflexivo, participativo, crítico, autocrítico, instructivo, flexible, integral y orientador” (Galarza López & Almuiñas Rivero, 2015), esto permite a cada institución de educación superior identificar que se requiere para que tengan un buen funcionamiento ante las entidades reguladoras y/o la normativas aplicadas a la educación superior; así como también, tener la capacidad para poder determinar la visión, misión que tendrá la institución ante la sociedad y al público en general.

A partir del plan estratégico de desarrollo institucional, se genera el plan operativo anual, el cual “incluira acciones pendientes e hitos que cada equipo o departamento debe llevar a cabo para ejecutar tu plan estratégico” (Martins, 2024), las distintas coordinaciones, áreas y/o departamentos tienen asignado las tareas, recursos e información que se debe presentar a la Unidad de Aseguramiento de la Calidad; el cual, revisa el cumplimiento a las peticiones impuesta por los organismos reguladores de las instituciones de educación superior. En el año 2010 Rafael Correa presidente de turno, a través del decreto 517 estableció que se fusionen entidades relacionadas al manejo de la educación, ciencia y tecnología de allí nacen al Consejo de Educación Superior, la Secretaría Nacional de Ciencia y Tecnología y el Consejo de Evaluación, Acreditación y Aseguramiento de la Calidad de la Educación Superior, cada uno con los roles que conllevaban a la supervisión, excelencia y establecer los mecanismo para asegurar la calidad en el ámbito de la educación superior (Pazmiño Moscoso, 2018). En este marco, la Ley Orgánica de Educación Superior, en su reforma de agosto del 2018, artículo 69, resalta la importancia de que se tenga un control en el aseguramiento de la calidad, destacando que la calidad en la enseñanza es indispensable en la formación superior (Asamblea Nacional República del Ecuador, 2018), consecuentemente se requieren tecnologías que permitan

Llevar un mejor control de las acciones y tareas asignadas a los distintas áreas que tiene la institución:

- Rectorado
- Vicerrectorado académico
- Coordinación académica
- Vinculación con la sociedad
- Bienestar estudiantil
- Aseguramiento de la calidad
- Educación continua
- Gestión tecnológica
- Marketing
- Investigación
- Sistemas
- Financiero

El AITEC a través del PEDI, se elabora el Plan Operativo Anual y los responsables de llevar la unidad de aseguramiento de la calidad en conjunto con las distintas áreas pertenecientes al instituto, se reúnen en espacios distintos donde se definen las tareas que estarán vinculadas a su cargo, las acciones y metas que deberá realizar son registrados un archivo de Excel compartido posteriormente con dichas acciones establecidas, en conjunto con el documento oficial del POA en formato PDF, una vez que sea aprobado por el Órgano Colegiado Superior, estos planes son críticos para definir objetivos a corto plazo para elaborar estrategias que respondan ante las exigencias educativas, rendición de cuentas, y demandas cambiantes en habilidades y enfoques pedagógicos.

El seguimiento del POA se lo realiza a través de sesiones de trabajo donde se analizan las acciones realizadas por cada responsable, los cuales deben enviar un informe de las actividades impuestas; además tienen que soportar la información provista con evidencias, que tendrán que subir a una carpeta compartida y según las fechas de corte para solicitar dicha información:

- Informe 1 antes de finalizar julio
- Informe 2 antes de finalizar noviembre, y
- Informe 3 antes de finalizar marzo

Estos informes tienen la finalidad de poder retroalimentar de forma eficiente cada uno de los avances del POA y de esta forma se puedan retroalimentar de forma correcta las estrategias para el PEDI, además de velar por el cumplimiento de los criterios solicitados por el CACES (2021); el cual, corresponde a un “modelo de evaluación externa con fines de acreditación para los Institutos Superiores Técnicos y Tecnológicos (ISTT)”, el modelo está dividido en distintos criterios y subcriterios, donde se establecen los indicadores a cumplir los cuales pueden ser de ámbito cuantitativo y/o cualitativo. Algunos de los antecedentes que han contribuido a la adopción del POA en este contexto incluyen:

- Necesidad de Rendición de Cuentas, al ser un espacio para informar las acciones, justificar los recursos utilizados ante la comunidad educativa, la sociedad y el estado, por lo expuesto dicha información es solicitada de distintas áreas dentro de la institución y esto según las tareas asignadas para cada una; por lo tanto, es indispensable que la información provista sea accedida de forma ágil.
- Cambio en la demanda de habilidades y enfoques pedagógicos innovadores, se requiere que exista un seguimiento oportuno en la educación de los estudiantes, actividades y recursos compartidas, monitoreo de procesos de la institución, todo esto búsqueda de mejoras y correcciones en el ámbito académico.
- El POA es indispensable para el monitoreo de las actividades asignadas a los responsables de cada departamento y darles un seguimiento a los informes provistos por cada uno, para que de esta forma se pueda revisar las medidas de corrección necesaria en los procesos académicos y administrativos.

Para esto el POA, es una herramienta fundamental aplicada a distintas organizaciones; por el cual, lo encontramos en las instituciones de educación en el ámbito superior, ya que permite a las instituciones, definir sus objetivos a corto

plazo y diseñar estrategias concretas para alcanzarlos durante un año académico, de aquí nacen algunas estrategias aplicadas al Plan Estratégico de Desarrollo Institucional.

## 1.2 DETERMINACIÓN DEL PROBLEMA

### 1.2.1 DESCRIPCIÓN DEL PROBLEMA

El plan operativo anual, es utilizado para el control y gestión a corto plazo, se derivan del PEDI, donde se obtienen las metas, objetivos operativos y acciones a seguir en cada departamento o área que tiene el AITEC, la descripción de estas actividades conlleva al alcance de los objetivos estratégicos planteados en el PEDI.

El AITEC tiene una unidad para el aseguramiento de la calidad, esta se encarga de la asignación de tareas; así como también el seguimiento de las distintas actividades establecidas a través de la entrega de informes al final de ciertos periodos, en ese instante se conocen las novedades que puedan existir en cada área, recopila la información a través de la recepción de informes por email, además de archivos compartidos que sustentan los informes entregados, posteriormente revisan y contrastan los archivos dados, esto requiere de mucho tiempo en la revisión, el deficiente o inadecuado manejo del Plan Operativo Anual -POA- puede acarrear una serie de problemas e implica a “repercusiones significativas, puesto que la falta de comprensión acerca de estos instrumentos de gestión podría afectar negativamente la participación, el compromiso y la colaboración de los diversos usuarios en los procesos de toma de decisiones y ejecución de acciones estratégicas” (Murillo Valverde, 2024), por lo expuesto mencionaremos algunos aspectos que se deberían considerar tales como:

- **Desalineación estratégica**, es importante que las metas y objetivos planificados estén correctamente estructurados y estos tienen mucha relevancia en el manejo de la información porque podrían no alinearse correctamente al PEDI y conlleven a un mal uso de recursos ya sean financieros, humanos y materiales. (Rojas Aquino et al., 2024)

- **Dificultades en la rendición de cuentas**, si no se lleva un seguimiento más eficiente en el POA, puede dificultar la rendición de cuentas; por tanto, se requiere de un monitoreo y control del mismo, se deben identificar quién es responsable del cumplimiento de la actividad; así como también, la documentación para sustentar cada aspecto tratado. (Bryson, 2011)
- **Riesgo de pérdida de acreditación**, el área que asegura la calidad de la educación requiere llevar un monitoreo adecuado en las actividades establecidas en el POA; con la finalidad de poder cumplir con los aspectos solicitados por los entes reguladores que vigilan a las instituciones y no incidir en la pérdida de la acreditación para la formación de profesiones (Barcia Cedeño et al., 2024)
- **Información desactualizada**, otro aspecto interno que puede causar un seguimiento fallido en las actividades del POA, es el no tener información actualizada sobre el progreso de las actividades planificadas, la dirección de la institución puede enfrentar dificultades para tomar decisiones oportunas para corregir el rumbo de las acciones propuesta si es necesario. (Menjívar Valencia, 2023)

No llevar un correcto seguimiento al POA podría causar problemas para el logro de las estrategias establecidas en el PEDI, las acciones y metas que tiene cada departamento son indispensables para que se alcancen los objetivos para la institución, además de que contribuyen al cumplimiento de las normativas vigentes por los entes reguladores de la educación superior.

## 1.2.2 FORMULACIÓN DEL PROBLEMA

¿Cómo implementar una solución tecnológica que facilite el seguimiento y control de las acciones y metas establecidas en el Plan Operativo Anual del Instituto Superior Universitario Almirante Illingworth?

### 1.2.3 JUSTIFICACIÓN DEL PROBLEMA

El CACES (2021) establece que una “planificación estratégica implica la formulación de estrategias que brinden una visión holística de la institución”; por tanto, el cumplimiento de las metas y objetivos planteado por la institución, requiere que sean integrados de manera ágil, para esto una herramienta tecnológica permitirá el registro de las acciones que se están realizando, indicando el avance de cada actividad y la visualización de todas las tareas asignadas, el monitoreo de los objetivos de forma eficaz pueden contribuir que los procesos que tiene la institución conlleven a tener una educación de calidad e inclusive exigido en la creación de universidades e institutos en la educación superior, artículo 109 numeral 2 de las LOES.

Cada institución de educación superior establece de qué manera se llevará el control y el monitoreo de las acciones que se realizará por parte de los entes responsables de que se cumplan estas actividades, el POA como ya se ha indicado posteriormente en este documento, la evaluación de sus resultados son a corto plazo y estas se van ajustando a los requerimientos del instituto y su entorno; para esto la Unidad de Aseguramiento de la Calidad, establece objetivos estratégicos, acciones y metas a realizar por cada área; así como también los medios de verificación, a continuación podremos observar los aspectos que trata el POA:

*Tabla 1.*

*Plan Operativo Anual*

<b>Objetivos estratégicos</b>	<b>Acciones</b>	<b>Metas y medios de verificación</b>
Fortalecer la formación técnico tecnológica en las modalidades semipresencial, híbrida y en línea del Instituto, atendiendo los	<ul style="list-style-type: none"> <li>• Completar programas</li> <li>• Elaborar guías</li> <li>• Formación practica</li> <li>• Servicios de biblioteca</li> <li>• Seguimiento y evaluación del proceso académico</li> </ul>	<ul style="list-style-type: none"> <li>• Programas</li> <li>• Informes</li> <li>• Proyectos</li> <li>• Certificados</li> <li>• Convenios</li> </ul>

Objetivos estratégicos	Acciones	Metas y medios de verificación
requerimientos del entorno local y regional.	<ul style="list-style-type: none"> <li>• Inserción de prácticas laborales</li> <li>• Fortalecimiento de recursos visuales</li> <li>• Estadísticas de uso de la plataforma</li> <li>• Manejo de distributivos</li> <li>• Encuentro de graduados</li> <li>• Ajustes curriculares</li> <li>• Procesos de titulación</li> </ul>	
Gestionar los procesos de investigación formativa y académico científico, favoreciendo la cultura científica y el desarrollo de programas y proyectos, que generen resultados para la difusión y transferencia a los diferentes campos de investigación desde un enfoque colaborativo.	<ul style="list-style-type: none"> <li>• Control del sistema de investigación</li> <li>• Control y seguimiento de la RED-DES</li> <li>• Control de proyectos de investigación</li> <li>• Publicaciones académicas</li> <li>• Eventos científicos</li> </ul>	<ul style="list-style-type: none"> <li>• Programas</li> <li>• Informes</li> <li>• Proyectos</li> <li>• Certificados</li> <li>• Convenios</li> </ul>
Potenciar la vinculación con la sociedad, mediante la implementación de programas y proyectos que favorezcan al desarrollo humano integral y logren transformaciones en el entorno.	<ul style="list-style-type: none"> <li>• Plan de educación continua</li> <li>• Actividades sociales y culturales</li> <li>• Convenios con entidades públicas y privadas</li> <li>• Diseño de programas</li> </ul>	<ul style="list-style-type: none"> <li>• Programas</li> <li>• Informes</li> <li>• Proyectos</li> <li>• Certificados</li> <li>• Convenios</li> </ul>

Objetivos estratégicos	Acciones	Metas y medios de verificación
Impulsar la gestión efectiva en todos los niveles del Instituto.	<ul style="list-style-type: none"> <li>• Control y seguimiento del POA</li> <li>• Planes de formación académica</li> <li>• Proceso de selección docente</li> <li>• Seguridad y salud ocupacional</li> <li>• Plan de marketing</li> <li>• Información financiera</li> <li>• Acceso a redes</li> </ul>	<ul style="list-style-type: none"> <li>• Programas</li> <li>• Informes</li> <li>• Proyectos</li> <li>• Certificados</li> <li>• Convenios</li> </ul>

*Nota: En la tabla se puede observar de forma concreta los objetivo, acciones metas y medios de verificación que tiene el POA, se debe notar que pueden cambiar en relación a las necesidades que se puedan suscitar. (Instituto Superior Almirante Illingworth, 2023)*

En la tabla anterior los objetivos trazados, se las da a conocer a cada área y estas según corresponda las acciones que deben desarrollar, la información se lleva a través de archivos digitales, las áreas involucradas remiten informes que son enviados vía correos electrónicos; además, se las debe subir en las carpetas compartidas de Google Drive; consecuentemente, se dividen los procesos y se determinan los procedimientos que siguen para la presentación de información, ya sea de índole administrativo o académico.

## 1.2.4 DELIMITACIÓN DEL PROBLEMA

El Plan Operativo Anual, necesita llevar un registro detallado de las tareas, metas, objetivos asignadas a cada área involucrada en los procesos administrativos del instituto y compartir medios de verificación que contrasten con la emisión de informes por cada responsable, un entorno donde se integre la información de tal manera que su acceso sea más ágil y evitar la pérdida e inconsistencia que se puedan dar al momento de llevar un seguimiento y control de las mismas.

### 1.3 JUSTIFICACIÓN DEL PROBLEMA

El Tecnológico Superior Almirante Illingworth (AITEC), como institución de educación superior, tiene la responsabilidad de cumplir con las normativas que rigen el sector educativo en Ecuador. Por ello, el instituto cuenta con un Plan Estratégico de Desarrollo Institucional que sirve como marco rector para orientar sus acciones hacia el logro de objetivos institucionales a mediano y largo plazo; a partir de este plan, se elabora el Plan Operativo Anual, un instrumento fundamental que permite traducir las metas estratégicas en actividades concretas distribuidas entre las distintas áreas y departamentos de la institución.

Llevar una gestión eficiente del POA tiene un alto grado de importancia para el funcionamiento del AITEC, ya que constituye una herramienta estratégica derivada del PEDI, define, asigna y monitorea actividades claves en las distintas áreas institucionales, un POA ágil permite aprovechar la oportunidades de los cambios endógenos y realizar cambios inmediatos de ser necesario; busca que se garantice el uso óptimo de los recursos y el cumplimiento de las normas establecidas por los organismos que regulan el aseguramiento de la calidad, según el marco de la Ley Orgánica de Educación Superior (2018).

Sin embargo, en la actualidad, el proceso de gestión y seguimiento del POA en el AITEC presenta limitaciones significativas, la falta de un sistema tecnológico adecuado que permita centralizar, actualizar y controlar en tiempo real el estado de cada actividad genera demoras en la revisión de informes, dificultades en la coordinación interdepartamental y riesgos de desalineación con respecto al PEDI; además, el manejo manual de la información a través de archivos compartidos y correos electrónicos incrementa la probabilidad de errores, duplicados o pérdida de información crítica, lo que afecta negativamente la toma de decisiones institucionales y pone en peligro el cumplimiento de requisitos regulatorios esenciales para la acreditación institucional.

En consecuencia, es indispensable contar con soluciones tecnológicas que optimicen la gestión del POA, garantizando una gestión más eficiente, transparente

y precisa, esto no solo contribuirá a mantener la alineación estratégica y a mejorar la rendición de cuentas, sino que también asegurará el cumplimiento adecuado de las normativas exigidas por entidades como la Secretaría Nacional de Ciencia y Tecnología (Senacyt), el Consejo de Educación Superior (CES) y el Consejo de Evaluación, Acreditación y Aseguramiento de la Calidad de la Educación Superior (CEACES); por lo tanto, la implementación de una herramienta tecnológica que fortalezca el manejo del POA es fundamental para mejorar la eficiencia operativa y asegurar el cumplimiento efectivo de los objetivos institucionales del AITEC.

## 1.4 OBJETIVOS

### 1.4.1 OBJETIVO GENERAL

Desarrollar una aplicación web integral que facilite la planificación, registro, seguimiento y análisis de las actividades, para la toma de decisiones estratégicas que contribuirán al cumplimiento exitoso de los objetivos institucionales, mediante el uso de tecnologías web modernas y prácticas ágiles

### 1.4.2 OBJETIVO ESPECÍFICOS

- Estimar de manera precisa los requisitos funcionales y técnicos de la aplicación, asegurando una comprensión clara de las necesidades específicas del Instituto Superior Universitario Almirante Illingworth.
- Implementar iterativamente un prototipo funcional de la aplicación, permitiendo la retroalimentación constante para validar conceptos y adaptarse ágilmente a los requisitos emergentes.
- Adoptar prácticas ágiles de desarrollo de software durante todo el ciclo del proyecto, garantizando la flexibilidad necesaria para ajustarse a cambios y prioridades, y asegurando una alineación efectiva con los objetivos institucionales.

## 2. MARCO TEÓRICO REFERENCIAL

Actualmente, la planificación y el manejo eficiente de actividades, así como los recursos, son fundamentales para cumplir con los objetivos estratégicos. En este contexto, el uso de herramientas tecnológicas que permitan optimizar los procesos administrativos de las instituciones es importante para que se lleve el control de los requerimientos que se deben presentar al CACES.

El marco teórico sustentará el desarrollo de la aplicación que está orientada al manejo de las gestiones de las actividades y recursos asignados en el Plan Operativo Anual (POA) del Instituto Superior Universitario Almirante Illingworth, donde se abordarán conceptos clave como la planificación operativa en instituciones educativas, el uso de microservicios; además, del impacto de las metodologías ágiles.

Con este análisis, establecemos una base que respalde la viabilidad de este proyecto, resaltando su contribución a como de optimiza gestión operativa y el manejo idóneo de los recursos dentro del Instituto Superior Universitario Almirante Illingworth.

### 2.1 PLAN OPERATIVO ANUAL

Este proyecto está basado en el manejo y gestión del POA; al ser una herramienta que permite integrar las metas institucionales, la asignación idónea de los recursos y la organización de las actividades en todas las áreas involucradas dentro del instituto, su implementación facilita la preparación y evidencia, además de la toma de evidencias en los procesos administrativos y académicos. (Bonilla et al., 2018).

EL POA, al ser un documento de planificación que detalla las metas y recursos necesarios para la ejecución de las actividades impuestas dentro de una institución educativa en un período específico (Corona et al., 2015), este monitorea como se

Lleva a cabo la obtención de todas las estrategias definidas, como se obtendrán los objetivos basados en el PEDI a un nivel táctico y operativo.

Según Rodríguez y León (2017), el POA debe cumplir con ciertas características fundamentales:

- Debe estar alineado con los objetivos institucionales.
- Debe poder ajustarse a cambios internos o externos.
- Debe incluir indicadores de desempeño que permitan evaluar su efectividad.
- Debe distribuir los recursos de manera óptima para garantizar su ejecución.

Consecuentemente, es de gran importancia el POA para las instituciones de educación superior, en relación a que conlleva a facilitar la gestión académica y administrativa, el POA nos va a permitir:

- Fortalecer la transparencia en la rendición de cuentas.
- Ser más eficiente en la distribución de los recursos.
- Fortalecer los mecanismos para el monitoreo y evaluación de los procesos actuales.

Por lo tanto, el POA no solo constituye una herramienta administrativa, sino que también tiene un rol importante en la utilización eficiente de recursos, la evaluación de resultados y promover mejoras continuas en los procesos de la institución; su correcta elaboración y seguimiento garantizará el cumplimiento de las normativas nacionales, la rendición de cuentas y la transparencia de los procesos frente a los organismos reguladores; así como también de toda la comunidad educativa.

## 2.2 EL USO DE MICROSERVICIOS EN EL DESARROLLO DE APLICACIONES WEB

Las aplicaciones de tipo web han ido evolucionando significativamente, inicialmente el diseño de aplicaciones con el uso de arquitecturas monolíticas, los

cuales era sistemas que se encontraban empaquetados en una única pieza donde las funcionalidades del sistemas se desplegaban juntas (Newman, 2019), hacia las arquitecturas basadas en microservicios, este enfoque permite mejorar la escalabilidad, el mantenimiento y la resiliencia de las aplicaciones a través del uso de distintos servicios, cada uno realizando su propio proceso para que fue creado y comunicándose a través de mecanismo que viaje por la HTTP, con el uso de microservicios puede proporcionar beneficios clave en términos de modularidad, integración de servicios y gestión eficiente de los recursos asignados, al ser tan cambiantes los requisitos para la presentación de la acreditación, así como también, responder de forma más flexible a los requerimientos institucionales establecidos en el PEDI.

## 2.2.1 ARQUITECTURA MONOLÍTICA

La arquitectura monolítica durante mucho tiempo ha sido el enfoque predominante en el desarrollo de software, se caracteriza por la integración de los elementos de una aplicación tales como, la lógica del negocio, el acceso a los datos y la interfaz de usuario, en una única unidad desplegable, esta estructura ofrece ventajas en términos de simplicidad inicial y despliegue unificado, también presenta importantes limitaciones cuando las aplicaciones se vuelven más complejas y/o tienen un mayor tamaño (Guaman et al., 2018).

Uno de los principales inconvenientes de este enfoque es su fuerte acoplamiento interno, que impide modificar o escalar componentes de manera independiente, limitando así la adopción de prácticas modernas como la integración y despliegue continuo, y la escalabilidad horizontal (Jaramillo Acevedo et al., 2017). Además, la complejidad crece exponencialmente cuando múltiples equipos deben trabajar sobre el mismo conjunto de código, lo que puede generar conflictos y errores recurrentes (Negrete Rodríguez & Infante Martínez, 2022).

Villamar et al. (2015), realizaron una evaluación empírica en la que compararon un desarrollo con arquitectura monolítica y otro basado en microservicios, ambos implementados en entornos de nube, concluyeron que los microservicios ofrecen

una mayor eficiencia en escenarios de alta demanda, aunque requieren una infraestructura más compleja.

Pasar de una arquitectura monolítica a microservicios es a menudo un procedimiento que debe ser cuidadosamente planificado, Díaz et al. (2019) destacó la importancia de contar con infraestructura seguras y con modelos ágiles para implementar DevSecOps, especialmente en entornos donde la disponibilidad y la seguridad son fundamentales. En la misma línea, Moreno-De La Cruz et al. (2023) señalan que, si bien el modelo monolítico puede ser más sencillo para productos mínimos viables, su transición hacia esquemas distribuidos resulta inevitable para lograr soluciones escalables.

Por tanto, la arquitectura monolítica sigue siendo muy útil al momento del diseño de aplicaciones o sistemas pequeños; sin embargo, no debe ser aplicado en entornos empresariales donde la flexibilidad y la adaptabilidad son elementos clave para el éxito del software.

## 2.2.2 MICROSERVICIO

Los microservicios representan un estilo arquitectónico donde la aplicación tiene varios servicios desacoplados y trabajan de forma independiente, los cuales se encarga de una funcionalidad o proceso específico y se comunica con protocolos HTTP/HTTPS (Di Meglio & Starace, 2024) esta independencia no permite que exista interferencia entre los servicios que se hayan creado, las principales características de los microservicios incluyen:

- Cada servicio opera de manera independiente, el desarrollo, despliegue y escala de cada componente no afecta a los demás.
- Es la capacidad de escalar servicios específicos según la demanda sin necesidad de escalar la aplicación completa.
- Cada microservicio puede ser desarrollado con diferentes tecnologías y bases de datos, facilitando la integración de diversos lenguajes y herramientas.

- El aislamiento de fallos para evitar que un error en un componente afecte a toda la aplicación.

Por lo expuesto anteriormente, podemos observar que el uso de microservicios, en comparación con el desarrollo tradicional basado en arquitecturas monolíticas, es un proceso que mejora la escalabilidad y mantenibilidad de las aplicaciones web gracias a su independencia funcional, a continuación, podremos observar en la tabla una comparación entre ambas arquitecturas

Tabla 2.

Comparación entre las arquitecturas monolíticas y microservicios

Característica	Monolítica	Microservicios
<b>Escalabilidad</b>	Escalado en bloque, costoso y complejo	Escalado por servicio, más eficiente
<b>Mantenibilidad</b>	Difícil de actualizar sin afectar al sistema	Permite cambios en servicios individuales
<b>Despliegue</b>	Lento, requiere redistribuir toda la aplicación	Rápido, cada servicio puede desplegarse por separado
<b>Tolerancia a fallos</b>	Un fallo puede afectar toda la aplicación	Aislado, un fallo no interrumpe otros servicios
<b>Lenguaje de Programación</b>	Un solo lenguaje para toda la aplicación	Puede usar diferentes lenguajes por servicio

*Nota: La tabla muestra cómo se comparan entre las principales características que tienen las arquitecturas de software (Fritzsich et al., 2019)*

En la tabla se puede apreciar que la arquitectura de microservicios ofrece ventajas significativas en comparación con la arquitectura monolítica, especialmente en términos de escalabilidad y tolerancia a fallos, es importante destacar que la capacidad de desplegar y actualizar servicios de manera independiente reduce la complejidad en el mantenimiento de cualquier sistema o aplicación, esto conlleva a que permita una rápida adaptación a nuevas necesidades del usuario, negocio o según sea el contexto del desarrollo.

En contraste, la arquitectura monolítica, aunque es más sencilla de implementar inicialmente, presenta desafíos en entornos dinámicos donde la flexibilidad y el escalado eficiente son fundamentales para evitar algún tipo de latencia, fallo en la seguridad al momento de realizar la comunicación y/o cualquier interacción que pueda existir entre los servicios diseñados; en consecuencia, la adopción de microservicios se vuelve un factor clave en el desarrollo de aplicaciones modernas, permitiendo mayor resiliencia y optimización de recursos (Mathews, 2025).

### 2.2.3 DESAFÍOS Y CONSIDERACIONES EN LA IMPLEMENTACIÓN DE MICROSERVICIOS

Los microservicios, tiene algunas ventajas al momento de realizar su implementación; sin embargo, también se deben considerar que presenta algunos otros desafíos que deben ser acogidos adecuadamente para que se cumpla de forma correcta la operatividad y funcionalidad de la aplicación.

Uno de los principales retos es la complejidad en la comunicación, debido a que los microservicios operan de manera independiente y estas requieren de mecanismos eficientes para la interacción entre ellos, en este sentido, es fundamental que para la interacción entre ellos se utilice API Gateway o sistemas de mensajería basada en eventos, dado que se optimizaría la comunicación y se podría evitar cuellos de botella y no se tenga problemas en el rendimiento de la aplicación en la que se esté desarrollando (Alshuqayran et al., 2016).

Adicionalmente, se debe tener en cuenta la seguridad que deben tener los microservicios, en el caso de no tomar las medidas necesarias para su protección podría exponerse en caso de que la aplicación sea utilizada por algún tipo de red sin filtros de seguridad, aumentando el riesgo de ataques, para esto es necesario implementar protocolos para la autorización y autenticación como OAuth2 y JSON Web Tokens, estas herramientas permiten controlar el acceso y garantizar la integridad de la información compartida entre servicios (Aristizabal et al., 2024).

Además, el monitoreo y depuración en arquitecturas de microservicios es más compleja en comparación con los sistemas monolíticos, ya que cada servicio genera múltiples registros y métricas, estos desafíos conllevan a una planificación estratégica en la adopción de microservicios, asegurando una infraestructura que optimice la comunicación, fortalezca la seguridad y garantice un monitoreo efectivo de la aplicación que se esté desarrollando.

## 2.3 METODOLOGÍAS ÁGILES EN EL DESARROLLO DE SOFTWARE

Las metodologías ágiles, son prácticas que permiten establecer entregables y un desarrollo de avance secuencial a través de un flujo de trabajo continuo, donde cada avance se irá retroalimentado con las acciones o fases que se vayan desarrollando, es decir que puede ir adaptándose en colaboración con el cliente (Canosa Ferreiro, 2024).

Se tienen algunos tipos de metodologías, cada uno con su propio enfoque en el desarrollo de software:

- SCRUM, el cual busca establecer entregables, cada uno definido en un intervalo de tiempo y priorizadas para a la necesidad del cliente.
- KANBAN, en este enfoque el equipo de trabajo del desarrollo define un flujo de trabajo a seguir, donde puede tener un estado pendiente, en progreso o terminado.
- XP, al igual que los otros tipos de metodologías tiene como enfoque establecer entregables cortos e incrementables, que permita la obtención del desarrollo planteado para el cliente.

Todas estas metodologías tienen en común principios fundamentales, la iteración de retroalimentación continua, la cooperación activa entre equipos y la adaptabilidad ante los cambios, estos aspectos las convierten en enfoques eficaces para el manejo de proyectos de software para contextos con un enfoque muy dinámicos y centrados en el usuario (Silva et al., 2024).

## 2.3.1 MANIFIESTO ÁGIL

El manifiesto ágil, se enfoca en cómo se realizará la colaboración del cliente con el desarrollo del producto y los cambios que se van dando a través del crecimiento del producto, está formado por cuatro valores y 12 principios (Canosa Ferreiro, 2024, pag 186)

- Primer valor, las personas son lo mas importantes, en segundo plano quedan las herramientas y los procesos
- Segundo valor, cada incremento es funcional y puede ser utilizados por lo clientes
- Tercer valor, fácil adaptación al cambio, aunque en el contrato con el cliente se haya establecido algo especifico
- Cuarto valor, busca general valor en los entornos VUCA los cuales representan un entorno cambiante

Cada valor está alineado a la comprensión y el manejo de que los proyectos sean con incrementos funcionales, que sean adaptables y den respuestas a los entornos cambiantes que están sujeto los proyectos, se tienen 12 principios (Beck et al., 2001)

1. Entrega continua y la participación del cliente para la retroalimentación y adaptación de cada iteración
2. Los procesos agiles aprovecha los cambios que se den en el proyecto de desarrollo
3. Se enfatiza a la entrega funcional y de forma continua del producto
4. El product owner y los desarrolladores trabajan de forma conjunta
5. Se busca que el equipo cree un producto de calidad para el cliente
6. Reuniones periódicas para una mejor comunicación
7. Se debe ir probando el software para revisar y evitar errores
8. Los desarrolladores o equipo de trabajo son capaces de entregar y adaptar de ser necesario el software sin que este lo sobrecargue
9. El producto debe mostrar agilidad a corto, mediano o largo plazo

10. Se busca eliminar el trabajo innecesario, automatizar tareas manuales y usos de librerías
11. Equipos autoorganizados para un mejor desarrollo del proyecto
12. El equipo de desarrollo hace referencia a la entrega continua

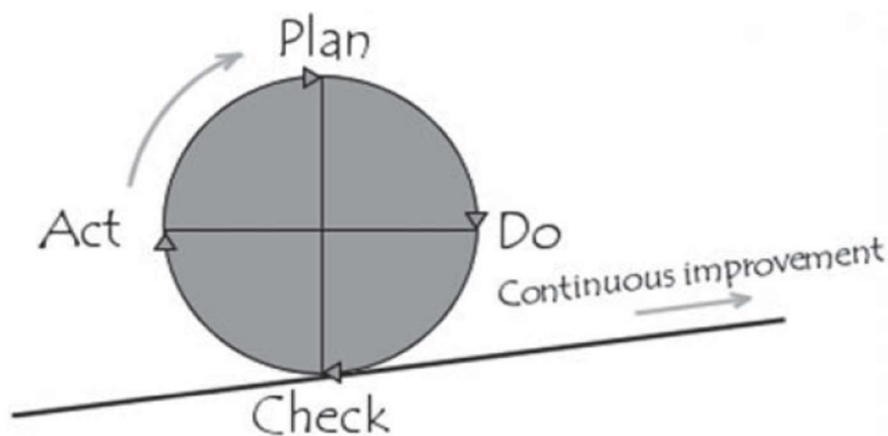
Para que el proyecto utilice scrum, debe cumplir con el manifiesto ágil y su filosofía de tal manera que el proyecto acoja la entrega continua del proyecto, se debe tener claro cómo serán los roles de cada participante en el proyecto, debe tenerse una estrecha relación entre el cliente y las personas o empresa que ese encargado del desarrollo de software.

### 2.3.2 SCRUM

Según Monte Josep (2016), scrum tiene un marco de trabajo en el desarrollo que propone roles, artefactos y actividades; además, busca como establecer una eficiente comunicación para que de esta forma la retroalimentación sea más eficaz en el diseño de la aplicación, está basado en el modelo del ciclo Deming, el cual lo podemos apreciar en la ilustración siguiente:

*Ilustración 1.*

*Ciclo de Deming (Plan-Do-Check-Act)*



*Nota: Esta herramienta es utilizada para la gestión y optimización de los procesos que andan en búsqueda de la mejora continua (Monte Galiano, 2016, pag. 22)*

Ambos se caracterizan por la iteración y procesos de mejora continua, donde cada aspecto del ciclo se alinea con las tareas realizadas en scrum, claro está que se tiene

ciertas diferencias entra ambos, dado a que el ciclo de Deming se puede desenvolver en cualquier área, por el contrario Scrum es exclusivamente para el desarrollo de software en proyecto ágiles, en cada marco del modelo propuesto por el ciclo de Deming, tiene ciertas bases en Scrum, podremos observarlo en la tabla siguiente.

Tabla 3.

Ciclo de Deming y Scrum

Ciclo de Deming	Scrum
Planificar, donde identifica los objetivos y las mejoras a realizar	Sprint planning, donde se definirá lo que se realizará en el srpint
Hacer, las soluciones que se vayan a implementar	Sprint, desarrollo incremental del producto a realizar
Verificar, se evalúan los resultados obtenidos	Sprint review, se busca identificar y aplicar mejoras
Actuar, búsqueda de estandarizar las mejoras que se vayan aplicar	Backlog Refinement, en este espacio se modifica de ser necesario un proceso para el siguiente sprint

*Nota: En esta tabla se observan cada aspecto que aborda el ciclo de Deming (Pinnington & Mir , 2014) con el marco de Scrum y como funciona cada uno de proceso de mejora (Schwaber & Sutherland, 2020)*

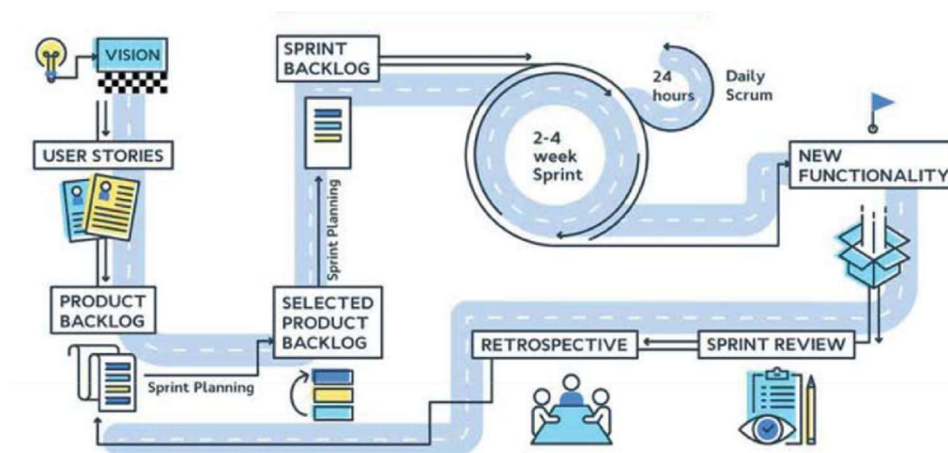
De la tabla anterior, lo que se busca comprender es que scrum esta basado en la búsqueda de realizar el desarrollo de un software de manera continua y cada interacción sirve para retroalimentar al misma o en la siguiente a través de sus sprint, los cuales son los incrementos secuenciales que se van dando en el proyecto de software, lo importante de resaltar de cada iteración o sprint es que deben completarse para poder seguir al siguiente sprint, esto conlleva a que el proyecto se vaya desarrollando constantemente y se vaya llevando un control de los acciones que se hayan planteado. (Canosa Ferreiro, 2024, pag 158).

El producto owner quien es el propietario del producto, en nuestro caso el que solicita el desarrollo de la aplicación o software, en el actor que solicita las prioridades de tareas en asignación de los sprint propuestos, debemos considerar que en la tabla anterior podemos ver que el primer aspecto a abordar es la

planificación del sprint o sprint planning, donde el cliente establece las prioridades y posteriormente de determinan las tareas a realizar, las cuales pueden darse en algunas reuniones con el equipo de trabajo del proyecto.

Ilustración 2.

Procesos de Scrum



Nota: La ilustración muestra como Scrum tiene un enfoque encargado de la creación de artefactos (Canosa Ferreiro, 2024, pag 206)

Al final de cada proceso del sprint debe entregarse un MVP, el cual corresponde a tener un producto mínimo viable, para el cual es importante como observamos en la figura se comienza con el product backlog, el cual se establece una lista de las peticiones por parte del cliente, estas tareas a realizar serán las historias de usuario, los cuales van desencadenando los sprints requeridos para el proyecto, estos generan los MVP y se crean las nuevas funcionalidades o retroalimentaciones que pueda tener el proyecto, los cuales pueden generar nuevas mejoras y permite alimentar al siguiente sprint o realizar las adaptaciones del sprint actual.

## 2.4 INTEGRACIÓN CONTINUA Y DESPLIEGUE CONTINUO (CI/CD)

## 2.5 DEFINICIÓN Y CONCEPTO DE CI/CD

Actualmente se han implementado prácticas en el desarrollo de software, estas buscan minimizar los errores; por lo tanto la Integración Continua (CI), se da cuando un equipo o una persona que se encuentra desarrollando alguna aplicación

especifica en algún proyecto y se utilice alguna plataforma donde conlleve a que su código se integre frecuentemente su codificación a la versión del proyecto de software principal, para esto, pasa por una revisión previa antes de que la plataforma en la que se encuentre, para que una versión del desarrollo en el que se encuentre sea considerado como un envío exitoso, esta debe estar sin errores, con esta prevención de errores se busca optimizar el tiempo de desarrollo del proyecto (Team, 2024), con la integración continua da como bases para la Implementación o Despliegue Continuo (CD); sin embargo has ciertos aspectos que se podrían considerar al momento de realizar la implementación continua:

- Analizar la arquitectura de del sistema, dado a que la aplicación actual está basada en microservicios, y esto conlleva a que cada uno se implementa de forma independiente, esto implica a elegir la mejor herramienta de implementación.
- Además, se debe determinar el entorno de implementación debido a que l ser una aplicación de microservicios, esta debe tener la capacidad para poder implementarse en diferentes clústeres, Adochiei et al. (2019) indica que son aquellos que permiten el despliegue de aplicaciones, al ser servidores interconectados, el clúster permite la administración y orquestación, balanceo, para que se asegure el correcto funcionamiento en el despliegue de las aplicaciones.
- Establecer cómo se realizarán los despliegues de la aplicación, considerando que estén definidos de forma correcta la ejecución de pruebas, además su entorno de producción.

Todo lo anterior, conlleva a que CI/CD sea una práctica importante para tener un buen desarrollo para la aplicación web, asegurando que la calidad y velocidad en el despliegue del desarrollo y este sea más eficiente.

## 2.6 IMPORTANCIA DE CI/CD EN EL DESARROLLO DE APLICACIONES WEB

Al momento que se adopte el uso de CI/CD, esto mejora la entrega de la aplicación y conlleva a que se controle cada aspecto del software que se está desarrollando, debido a que se logra una mejor escalabilidad en el desarrollo, lo que responde a las exigencias y cambios que puedan surgir en las solicitudes de evaluación realizadas por el CES; así como también, a los ajustes que puedan presentarse en el PEDI al contar con una implementación y despliegue más eficientes se evitarían problemas en la publicación del software (Mbuta, 2024).

Por lo antes mencionado en aplicaciones de gestión tales como el POA, CI/CD permite que nuevas funcionalidades sean desplegadas sin afectar el uso de la aplicación a los usuarios, optimiza el flujo de trabajo entre los desarrolladores, promueve el trabajo colaborativo, mejorando el ciclo del desarrollo del programa (Chippagiri & Ravula, 2024).

## 2.7 HERRAMIENTAS Y TECNOLOGÍAS

Para abordar estos desafíos, se utilizarán tecnologías tales como:

- ReactJS, Según Meta (2023), “React permite construir interfaces de usuario mediante componentes reutilizables que manejan su propio estado y se actualizan eficientemente” (p. 12)., al tener un enfoque declarativo en el desarrollo de las interfaces y a través del uso de bibliotecas de javascript, estas librerías permiten que las interfaces web tengan un sentido más dinámico; en relación al tener un mejor manejo en el DOM del sitio, las mismas están asociados un entorno de bibliotecas virtuales que maneja React permitiendo un acceso más ágil de cada uno de ellos en la creación de las interfaces. (Jimenez Villar, 2016), cada componente puede ser reutilizable y tener un excelente renderizado, optimizando la carga de las páginas y el tiempo de respuesta de la aplicación (Veeranjaneyulu Veeri, 2024).

- PostgreSQL, se caracteriza porque puede manejar datos en el estándar SQL, matrices, JSON, además de un buen manejo en el control de concurrencias múltiples, además tiene una eficiencia en la simultaneidad del acceso a los datos; así como también, baja latencia, el cual es el tiempo de respuesta en la obtención de los datos (Salunke & Ouda, 2024).
- Github, la plataforma tiene un enfoque colaborativo y se lo utilizará en el control de versionamiento del proyecto, además a través de esta herramienta se llevara a cabo el CI/CD, utilizando las herramientas de Git Actions que permiten un análisis del código, hacer pruebas, compilación, despliegues esto permite tener un código más limpio; sin embargo, Github, también puede interactuar con herramientas externas Jenkins, Travis CI, CircleCI y Docker u otras que conlleven a pruebas automatizadas, despliegue continuo, un control de versiones estando dentro del mismo contexto de GitHub (Saavedra et al., 2024)
- NGINX, es el servidor web que se utilizará para la aplicación web, también trabaja en algunas aplicaciones como proxy inverso, es de código abierto y de tipo asíncrono, adicionalmente al estar orientado a eventos el servidor mantiene un consumo estable de memoria y tiene una mejor respuesta ante las conexiones concurrentes (Kunda et al., 2017).

Consecuentemente, es importante destacar que con la ayuda de estas herramientas permitan cumplir con los aspectos requeridos.

## 3. MATERIALES Y METODOLOGÍA

### 3.1 ENFOQUE METODOLÓGICO

En el contexto actual del desarrollo de software, las metodologías ágiles han adquirido relevancia debido a su capacidad para responder rápidamente a los cambios, fomentar la colaboración interdisciplinaria y entregar valor de manera continua, scrum se destaca por su estructura clara, iterativa e incremental, lo que lo convierte en una opción ideal para proyectos orientados al desarrollo de sistemas institucionales, como es el caso del proyecto propuesto para el control del Plan Operativo Anual (POA) del Instituto Superior Universitario Almirante Illingworth (AITEC).

Diversas investigaciones coinciden en que scrum se distingue por su estructura bien definida, que incluye roles específicos como Scrum Master, Product Owner y el equipo de desarrollo, así como iteraciones planificadas denominadas sprints y reuniones regulares, conocidas en el ámbito de Scrum como daily meetings, esto facilita una planificación organizada y una alta adaptabilidad ante los cambios; Según Quelal et al. (2018), scrum fue adoptado por el 68.75% de los equipos analizados en Ecuador, superando significativamente a Kanban (25%) y XP (18.75%), lo que evidencia su alta aceptación en contextos reales de desarrollo de software.

En una revisión sistemática de 925 papers Hron y Obwegeser (2022), en sus análisis concluyeron que scrum es una metodología muy versátil y es adoptado en variedad de entornos tales como organizaciones educativas, gubernamentales y empresariales, concluyen que esta metodología no solo mejora la comunicación interna y la adaptabilidad ante cambios, sino que también incrementa de manera significativa la probabilidad de éxito del proyecto; sin embargo no se tiene que evidenciar el uso de buenas prácticas en el desarrollo de software.

En la tabla siguiente se observa un análisis comparativo que presenta una síntesis de las principales características de las metodologías ágiles más utilizadas

Tabla 4.

Características Generales de las Metodologías Ágiles

Metodología	Enfoque principal	Iteraciones	Roles definidos	Adaptabilidad al cambio	Complejidad
<b>Scrum</b>	Gestión de proyectos mediante sprints	Sí (sprints de 2-4 semanas)	Sí (Product Owner, Scrum Master, Development Team)	Alta	Baja-Media
<b>Kanban</b>	Visualización del flujo de trabajo	No (flujo continuo)	No	Alta	Baja
<b>XP (Extreme Programming)</b>	Prácticas técnicas avanzadas	Sí (iteraciones cortas)	No	Alta	Alta

Nota: Este análisis comparativo presenta las principales características de las metodologías más utilizadas (Schwaber & Sutherland, 2020), (Hoda et al., 2012), (Rising & Janoff, 2000)

De acuerdo con el análisis comparativo mostrado en la tabla anterior, scrum se presenta como la metodología más equilibrada y funcional para el desarrollo de un sistema, gracias a su enfoque iterativo, roles bien definidos, alta adaptabilidad al cambio y bajo nivel de complejidad relativa, este modelo asegura una gestión eficiente del proyecto, scrum establece roles fundamentales:

Tabla 5.

Roles de Scrum en el AITEC

Rol Scrum	Quién lo desempeña en AITEC	Responsabilidad principal
<b>Product Owner</b>	Unidad de Aseguramiento de la Calidad	<ul style="list-style-type: none"> <li>• Representar las necesidades del cliente interno, es decir, de todos los departamentos involucrados en el POA.</li> <li>• Priorizar las funcionalidades del sistema de acuerdo con las</li> </ul>

Rol Scrum	Quién lo desempeña en AITEC	Responsabilidad principal
		necesidades institucionales y normativas. <ul style="list-style-type: none"> <li>• Mantener y actualizar el Product Backlog, verifica que se lleve con precisión los objetivos estratégicos del PEDI y metas del POA.</li> <li>• Validar los entregables al final de cada sprint, asegurando que cumplan con los estándares de calidad exigidos por organismos reguladores como el CEACES.</li> </ul>
<b>Scrum Master</b>	Líder del proyecto	Facilita las reuniones, elimina los impedimentos y fomenta la mejora continua.
<b>Development Team</b>	Líder del proyecto	Construye y entrega los incrementos del sistema POA en cada sprint.
<b>Stakeholders</b>	Departamentos funcionales del instituto	Proporcionan retroalimentación continua sobre los entregables, participan en las validaciones y colaboran en la identificación de prioridades.

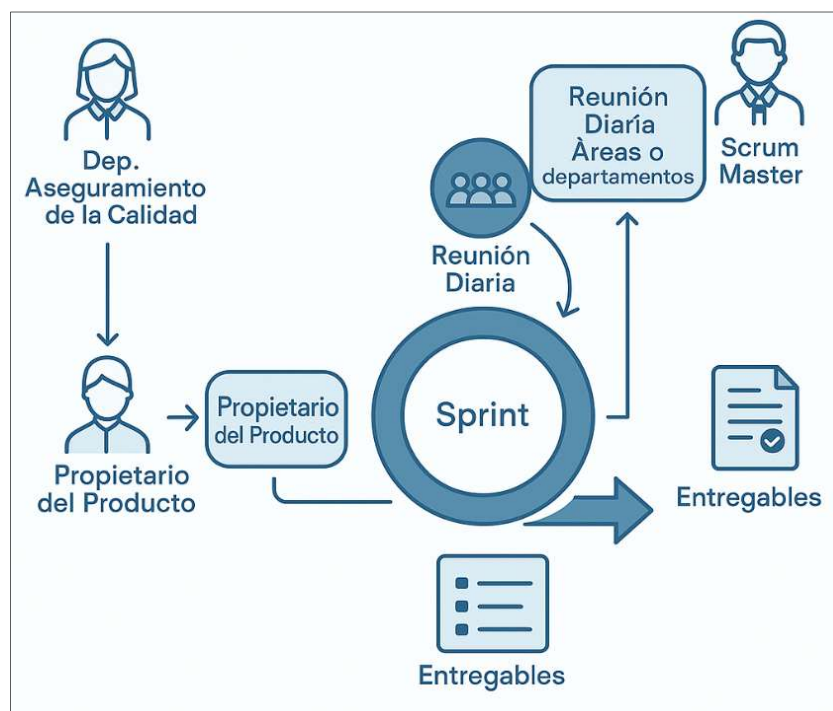
*Nota: Es importante recalcar que el líder del proyecto será el autor de esta propuesta.*

La tabla anterior muestra la estructura los roles que tendrán cada actor dentro del instituto; el cual, define claramente las responsabilidades y fomenta un mayor compromiso del equipo (Sutherland & Schwaber, 2020), esto es fundamental en un entorno institucional como el AITEC, donde diversas áreas que están inmersos en la gestión del POA.

Los sprints permiten entregar funcionalidades cada 2 a 4 semanas, lo cual garantiza retroalimentación constante, esta característica es fundamental en el desarrollo de la aplicación para la gestión del POA, realiza ajustes tempranos y valida funciones con los responsables del aseguramiento de calidad y adaptarse a nuevas normativas o requisitos cambiantes.

Ilustración 3.

Flujo Scrum



Nota: En la imagen se muestra cual es flujo que tiene la metodología aplicada

Con la imagen anterior, podemos observar el flujo que seguirá la propuesta de trabajo, al aplicar Scrum en la propuesta se podrá minimizar el riesgo de enfrentar problemas en el desarrollo del proyecto.

## 3.2 HERRAMIENTAS UTILIZADAS

El stack tecnológico para el diseño del proyecto, lo podremos observar en la tabla siguiente, donde encontraremos que tecnologías se utilizó y cuales fue el criterio para poder utilizarla en el desarrollo

Tabla 6.

Stack Tecnológico de la aplicación

Tecnología	Uso en el proyecto	Justificación
<b>React.js</b>	Front-end web	<ul style="list-style-type: none"> <li>• Se tiene experiencia en el desarrollo del uso de este framework</li> <li>• Es altamente modular, ideal para sitio web dinámicos</li> <li>• Tiene componentes reutilizables</li> <li>• Se integrará con las API REST de los microservicios utilizando Fetch API.</li> <li>• Uso de herramientas como Redux o Context API para la gestión global del estado de la aplicación.</li> </ul>
<b>Feathers.js</b>	Backend / API Gateway	<ul style="list-style-type: none"> <li>• Es un framework minimalista y flexible basado en Node.js ideal para construir API's</li> <li>• Se usará para crear cada uno de los microservicios:               <ul style="list-style-type: none"> <li>○ Autenticación</li> <li>○ Gestión de usuarios y roles</li> <li>○ Actividades del POA</li> <li>○ Objetivos estratégicos</li> <li>○ Informes y medios de verificación</li> </ul> </li> </ul>

Tecnología	Uso en el proyecto	Justificación
		<ul style="list-style-type: none"> <li>• Integración con bases de datos locales, en este caso PostgreSQL</li> <li>• Soporte para JWT para autenticación segura tal como la aplica con Outh2 para Google</li> </ul>
<b>PostgreSQL</b>	Base de datos única centralizada	<ul style="list-style-type: none"> <li>• Sistema robusto y escalable, ideal para datos estructurados como objetivos, metas y evidencias del POA.</li> <li>• Tiene un manejo eficiente en los recursos de la máquina</li> <li>• Menor costo operativo al ser open source</li> <li>• Cuenta con una comunidad activa para desarrolladores y se tienen un manejo completo de los datos y su infraestructura</li> </ul>
<b>Ubuntu Server</b>	Entorno de ejecución de la aplicación	<ul style="list-style-type: none"> <li>• Sistema operativo estable y seguro en servidores Linux, y al ser open source lo tenemos instalados en los servidores del instituto.</li> <li>• Es compatible con Docker, Nginx, Node.js y herramientas de CI/CD</li> </ul>
<b>Nginx</b>	Proxy reverso / balanceador de carga	<ul style="list-style-type: none"> <li>• Mejora rendimiento y seguridad. Redirige peticiones</li> </ul>

Tecnología	Uso en el proyecto	Justificación
		<p>al backend según reglas de configuración.</p> <ul style="list-style-type: none"> <li>• Actúa como proxy inverso para enrutar las solicitudes entrantes a los diferentes microservicios.</li> <li>• También puede funcionar como un balanceador de carga si se escala a múltiples instancias.</li> <li>• Optimiza el rendimiento del front-end estático en la compilación de React y protege contra amenazas comunes, como ataques DDoS y limitación de tasa.</li> </ul>
<b>GitHub</b>	Repositorio para control de versiones del código fuente	<ul style="list-style-type: none"> <li>• Permite un seguimiento de los cambios realizado en el proyecto, gestión de ramas</li> </ul>
<b>SonarCloud</b>	Análisis automático de calidad del código fuente del back y el front	<ul style="list-style-type: none"> <li>• Detecta bugs, vulnerabilidades, code smells y evalúa cobertura de pruebas en tiempo real en cada <i>push</i>.</li> </ul>
<b>GitHub Actions</b>	Automatización CI/CD	<ul style="list-style-type: none"> <li>• Ejecuta flujos automáticos para pruebas, análisis estático y despliegue en Docker al hacer commits o pull requests.</li> </ul>
<b>Docker</b>	Contenedores para desplegar	<ul style="list-style-type: none"> <li>• Facilita entornos reproducibles y portables. Cada microservicio</li> </ul>

Tecnología	Uso en el proyecto	Justificación
	microservicios y la base de datos	se despliega como contenedor independiente.
<b>Autenticación Google (OAuth2)</b>	Proveedor de identidad para acceso al sistema	<ul style="list-style-type: none"> <li>Mejora la seguridad, usando las cuentas institucionales de Google para acceder al sistema del instituto.</li> </ul>

*Nota: Las tecnologías aplicadas fueron alineado a la arquitectura de microservicios*

Con la tabla anterior, se exponen las ideas del porque se utilizaron en el desarrollo de la aplicación web.

### 3.3 LEVANTAMIENTO DE REQUERIMIENTOS

Este proyecto se basa en lo proporcionado por el POA del AITEC; en el cual, se detallan los desafíos operativos, estratégicos y normativos que enfrenta la institución, a partir de esta premisa, se identificaron los problemas y las limitantes en el manejo de la planificación, monitoreo y gestión de las actividades institucionales.

*Tabla 7.*

*Necesidades del POA del AITEC*

Necesidad	Justificación
<b>Sistema centralizado para el POA</b>	Evitar el uso de múltiples herramientas manuales y dispersas
<b>Autenticación segura con roles</b>	Garantizar acceso diferenciado según responsabilidades
<b>Registro de objetivos, acciones y metas</b>	Vincular cada actividad al PEDI para mantener alineación estratégica
<b>Seguimiento en tiempo real</b>	Monitorear el progreso de cada actividad y su cumplimiento

Necesidad	Justificación
<b>Carga de medios de verificación</b>	Documentar el avance con pruebas visibles y descargables
<b>Panel de control</b>	Visualización del estado del POA por departamento y objetivo
<b>Generación automática de informes</b>	Facilitar la rendición de cuentas y cumplimiento normativo
<b>Notificaciones y alertas</b>	Recordatorios de fechas límite y actualizaciones pendientes
<b>Respaldo automatizado</b>	Prevenir pérdida de datos y garantizar continuidad operativa

*Nota: La información mostrada nace a partir de los hallazgos encontrados en el proyecto desarrollado*

Por tanto, la identificación de las necesidades permitió establecer un conjunto claro de requisitos funcionales que guiarán el diseño e implementación de la solución destinada a mejorar la eficiencia operativa, la transparencia y el cumplimiento institucional:

*Tabla 8.*

*Requerimientos funcionales de la aplicación*

#	Requisito Funcional	Descripción	Prioridad	Tecnología Asociada
<b>RF-01</b>	Registro de usuarios con OAuth Google	Los usuarios sesionan a través del uso de sus cuentas institucionales de Google.	Alta	OAuth 2.0 / Firebase Admin SDK
<b>RF-02</b>	Asignación de roles por usuario	El sistema asigna roles como administrador, usuario o supervisor, según su función dentro del POA.	Alta	JWT + Feathers.js
<b>RF-03</b>	Gestión de departamentos	Permite registrar, editar, eliminar y listar los departamentos que participan en el POA.	Media	Feathers.js + PostgreSQL

#	Requisito Funcional	Descripción	Prioridad	Tecnología Asociada
RF-04	Creación de objetivos estratégicos	Cada objetivo estratégico debe estar vinculado al PEDI y tener asociadas acciones y metas.	Alta	Feathers.js + React
RF-05	Registro de acciones por departamento	Cada departamento debe registrar las acciones que realizará durante el año, incluyendo descripción, plazo y responsable	Alta	Microservicio de Actividades + PostgreSQL
RF-06	Seguimiento de metas y medios de verificación	Se debe registrar el avance de cada meta, permitiendo cargar documentos que sirvan como medio de verificación.	Alta	Frontend React + Backend Feathers.js + Almacenamiento en servidor
RF-07	Actualización del estado de actividades	Los responsables deben poder marcar actividades como: Pendiente, En progreso, Completado o Atrasado.	Alta	React + Feathers.js
RF-08	Generación de informes automáticos	El sistema genera una visualización de informes en formato PDF o XLSX con el estado actual del POA por departamento.	Alta	Report Service + React + Nginx
RF-09	Notificaciones y alertas de vencimiento	El sistema debe notificar a los responsables cuando se aproxime o venza el plazo de una actividad.	Media	Microservicio de Notificaciones
RF-10	Panel de control (Dashboard)	Mostrar un resumen visual del cumplimiento del POA por departamento y por objetivo estratégico.	Alta	React
RF-11	Búsqueda y filtrado de actividades	Permitir buscar y filtrar actividades por departamento,	Media	Feathers.js + Filtros en frontend

#	Requisito Funcional	Descripción	Prioridad	Tecnología Asociada
		responsable, estado, fecha límite, etc.		
<b>RF-12</b>	Historial de cambios	Registrar quién hizo modificaciones, qué cambió y cuándo se realizó.	Media	JWT + Logs en base de datos
<b>RF-13</b>	Respaldos automáticos de datos	Programar respaldos periódicos de la base de datos y archivos cargados.	Media	GitHub Actions + Docker
<b>RF-14</b>	Exportación de información	Permitir exportar datos del POA en formatos como CSV, Excel o PDF.	Media	Backend Feathers.js + Librerías de exportación
<b>RF-15</b>	Configuración de fechas límite por actividad	Asignar fechas de inicio y fin a cada actividad para facilitar el seguimiento.	Alta	React + Feathers.js
<b>RF-16</b>	Validación de entregas	Verificar que se hayan cargado todos los medios de verificación antes de marcar una meta como completada.	Alta	Validación backend + UI feedback
<b>RF-17</b>	Comentarios y colaboración entre áreas	Permitir comentarios en cada actividad para facilitar la coordinación interdepartamental.	Baja	Microservicio de comentarios + WebSockets
<b>RF-18</b>	Soporte multiusuario concurrente	Varias personas deben poder trabajar simultáneamente sin conflictos.	Alta	Arquitectura RESTful + Base de Datos relacional

*Nota: Los requerimientos fueron extraídos en bases a las necesidades encontradas*

Las prioridades que se asignaron a cada uno los requerimientos en la tabla anterior se lo describen de la siguiente manera:

- **Alta**, se lo clasifica como esencial para resolver los problemas identificados.
- **Media**, es importante pero no es crítico para el funcionamiento de la aplicación.

- **Baja**, son aquellas funcionalidades que pueden mejorar la experiencia de la aplicación; sin embargo, no son indispensables.

Adicionalmente, tenemos cuales son los requerimientos asociados según el problema planteado

Tabla 9.

*Requerimientos funcionales con las problemáticas planteadas*

<b>Problemáticas</b>	<b>Requisitos Funcionales Asociados</b>
<b>Desalineación estratégica</b>	RF-04, RF-05, RF-10
<b>Dificultades en la rendición de cuentas</b>	RF-02, RF-07, RF-12
<b>Riesgo de pérdida de acreditación</b>	RF-06, RF-08, RF-14
<b>Información desactualizada</b>	RF-07, RF-09, RF-11
<b>Pérdida de información crítica</b>	RF-13, RF-06
<b>Falta de integración entre áreas</b>	RF-17, RF-16
<b>Procesos manuales y lentos</b>	RF-01, RF-08, RF-10

*Nota: Los requerimientos fueron asociados según las problemáticas planteadas*

Con la tabla anterior se busca definir los requerimientos funcionales a los que cubrirá cada necesidad encontrada.

### 3.4 DISEÑO DEL SISTEMA

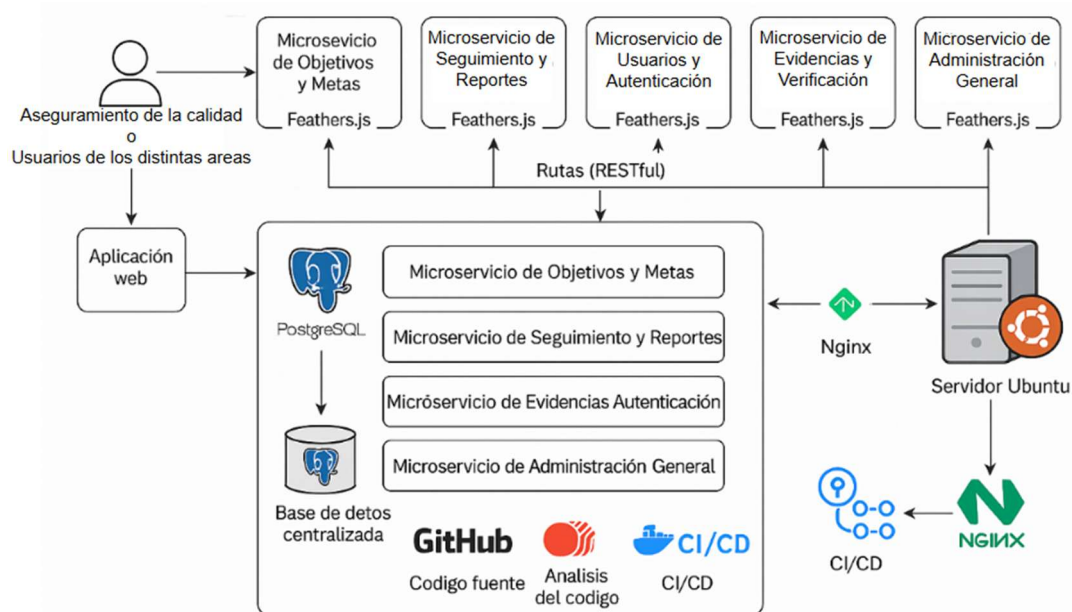
La aplicación que se diseñará tendrá un enfoque de tipo web, con una arquitectura de microservicios; por lo tanto, se buscará que la plataforma permita a la institución gestionar su Plan Operativo Anual, los usuarios podrán acceder a la plataforma a través de un navegador y contarán con diferentes roles y permisos de autenticación, según las responsabilidades en el proceso de planificación y ejecución del POA para cada área del instituto.

Kumar et al. (2022), menciona que “uno de los pasos clave en la construcción de un sistema basado en microservicios es identificar la división más efectiva de sistemas que, aunque se implementan individualmente, interactúan entre sí de manera cohesiva”; por tanto, con el objetivo de responder de manera eficaz los desafíos

institucionales relacionados con la gestión de las acciones planificadas, se divide la aplicación en servicios independientes, cada módulo que se defina, será el encargado del funcionamiento correcto en el proyecto, esto mejorará la escalabilidad, el mantenimiento y la flexibilidad en la evolución del software ante cualquier requerimiento o cambio que se suscite para el POA.

Ilustración 4.

Aplicación web del POA institucional



Nota: La imagen se muestra cómo se organiza la arquitectura del proyecto

Cada microservicio será desarrollado utilizando Feathers.js, un framework orientado a servicios RESTful, se crearán API's que serán modulares y reutilizables, en la imagen anterior podemos observar los microservicios previstos para la aplicación:

- **Microservicio de Objetivos y Metas**, administra la creación, edición y visualización de objetivos estratégicos y metas asignadas, .
- **Microservicio de Seguimiento y Reportes**, gestiona el avance de las acciones, generación de informes por departamento y alertas.
- **Microservicio de Usuarios y Autenticación**, gestiona usuarios, roles y la autenticación mediante cuentas institucionales de Google (OAuth2).

- **Microservicio de Evidencias y Verificación**, permite la carga y validación de medios de verificación documentos, archivos, certificados.
- **Microservicio de Administración General**, configuraciones del sistema, estructuras organizativas.

Todos estos servicios se comunican con una base de datos centralizada en PostgreSQL, aunque las tablas están lógicamente separadas para que se respete la autonomía por cada servicio, compartiendo el mismo motor que estará alojada junto al sistema en el servidor de Ubuntu, el cual se encuentra de forma local en el instituto.

El frontend del sistema esta desarrollado en React.js, lo que permitirá construir una interfaz web dinámica que facilite la interacción de los distintos departamentos con sus respectivas actividades en el POA, el frontend se conectará al backend a través del API Gateway, desarrollado con Feathers.js, y que dirigirá las solicitudes a los microservicios correspondientes.

La arquitectura contempla el uso de Docker para contenerizar cada microservicio, asegurando entornos aislados y portátiles para el despliegue del sistema, los contenedores serán los responsables a través de un sistema de orquestación; mientras que Nginx actuará como servidor proxy inverso, gestionando de manera segura y eficiente la redirección de peticiones al frontend y a los microservicios internos.

En lo que respecta al control de las versiones de la aplicación, el código fuente será gestionado a través de GitHub, y de igual forma también se integrará con SonarCloud para realizar análisis automáticos de la calidad del código, con el análisis se buscarán errores, vulnerabilidades y malas prácticas en cada push o pull request que se genere durante el desarrollo; además, se utilizará GitHub Actions como herramienta de automatización para ejecutar flujos de integración y despliegue continuo, asegurando que cada cambio pase por etapas de validación antes de ser liberado en producción hacia el servidor de Ubuntu que se encuentra de forma local en el instituto.

Adicionalmente, es importante recalcar que, para garantizar la seguridad y control de acceso, la aplicación integrará autenticación mediante cuentas de Google - OAuth2-, permitiendo a los usuarios institucionales acceder al sistema con sus credenciales corporativas asegurando así el inicio de sesión de forma segura.

### 3.5 IMPLEMENTACIÓN POR SPRINTS

Para este ítem se dispondrá de 4 horas diarias y solo se tendrán los días laborables que corresponde de lunes a viernes

Tabla 10

Distribución de Horas y duración total del Proyecto

Componente	Horas Estimadas	Días Laborales	Semanas Aprox.
<b>Infraestructura y Configuración Inicial</b>	20 hrs	5 días	1 semana
<b>Autenticación y Gestión de Usuarios</b>	24 hrs	6 días	1.2 semanas
<b>Objetivos Estratégicos</b>	20 hrs	5 días	1 semana
<b>Actividades y Metas</b>	28 hrs	7 días	1.4 semanas
<b>Medios de Verificación</b>	20 hrs	5 días	1 semana
<b>Generación de Informes</b>	24 hrs	6 días	1.2 semanas
<b>Frontend (React.js)</b>	32 hrs	8 días	1.6 semanas
<b>CI/CD y Automatización</b>	20 hrs	5 días	1 semana
<b>Documentación y Pruebas</b>	36 hrs	9 días	1.8 semanas
<b>Total</b>	<b>224 hrs</b>	<b>56 días</b>	<b>11,2 semanas</b>

*Nota: Tiempo aproximado para el desarrollo de la aplicación.*

Consecuentemente, con lo anterior podemos establecer cada sprint a realizar dentro del proyecto, permitiendo establecer el tiempo aproximado que se tomará para el desarrollo del proyecto.

## 3.6 SPRINT 1 – PREPARACIÓN TÉCNICA E INFRAESTRUCTURA

Este primer sprint se centra en configurar el entorno y se definen cuáles serían las herramientas necesarias para iniciar el proyecto, se crea un repositorio y se definen la estructura del proyecto; además se prepara el entorno local y el servidor de Ubuntu.

Tabla 11.

*Sprint 1, configuración inicial del proyecto*

ID	Actividad	Horas	Entregable
<b>INF-01</b>	Crear repositorio en GitHub	4	Repositorio inicializado
<b>INF-02</b>	Definir estructura base del proyecto	4	Carpeta src, .gitignore, README
<b>INF-03</b>	Instalar dependencias iniciales	4	Node.js, Feathers.js, Docker
<b>INF-04</b>	Configurar Docker Compose básico	4	Archivo docker-compose.yml funcional
<b>INF-05</b>	Preparar servidor Ubuntu	4	Docker, Nginx y PostgreSQL instalados

*Nota: Actividades correspondiente al sprint 1*

En este primer sprint se dejará listo para el uso y el desarrollo de la aplicación, para este caso

## 3.7 SPRINT 2 – AUTENTICACIÓN Y GESTIÓN DE USUARIOS

Este sprint se centra en desarrollar el microservicio de usuarios y autenticación segura mediante Google (OAuth2), incluyendo roles y JWT para sesiones activas, a la que estará expuesta nuestra aplicación

Tabla 12

Sprint 2, Diseño del microservicio de autenticación y sesión de usuarios

ID	Actividad	Horas	Entregable
<b>AUTH-01</b>	Diseñar modelo de usuarios y roles	4	Estructura de datos en PostgreSQL
<b>AUTH-02</b>	Implementar servicio de autenticación	6	Login con Google integrado
<b>AUTH-03</b>	Validar JWT en rutas protegidas	6	Middleware de autenticación
<b>AUTH-04</b>	CRUD básico de usuarios	4	Registro y edición de usuarios
<b>AUTH-05</b>	Asignar usuarios a departamentos	4	Relación usuario-departamento

*Nota: Se describen las acciones que se realizarán*

Para esto se construye a través de la instalación de las dependencias para cada proceso, se revisarán las sesiones de cada usuario y el rol que desempeña en la aplicación.

### 3.8 SPRINT 3 – GESTIÓN DE OBJETIVOS Y SEGUIMIENTO DE METAS

En este sprint se desarrollan los microservicios relacionados con los objetivos estratégicos del PEDI, como se registrarán las acciones asociadas y el seguimiento de metas, asegurando alineación con el plan institucional POA.

Tabla 13

Sprint 3, gestión de objetivos, metas y acciones

ID	Actividad	Horas	Entregable
<b>OBJ-01</b>	Diseñar modelo de objetivos	8	Base de datos actualizada

ID	Actividad	Horas	Entregable
<b>OBJ-02</b>	Microservicio de objetivos	12	API REST funcional
<b>ACT-01</b>	Diseñar modelo de acciones y metas	10	Esquema de actividades
<b>ACT-02</b>	Microservicio de seguimiento	10	Servicio CRUD de actividades
<b>ACT-03</b>	Notificaciones programadas	8	Alertas de vencimientos

*Nota: Diseño del microservicio para el seguimiento*

En esta apartado se acogieron dos aspectos de lo sprints planteado en la tabla donde se describe la distribución de horas del proyecto

### 3.9 SPRINT 4 – DESARROLLO DEL FRONT-END

Este sprint se enfoca a la construcción de la interfaz web con React.js, conectándola al backend y ofreciendo a los usuarios una experiencia visual clara e intuitiva.

*Tabla 14*

*Sprint 4, Diseño del sitio web*

ID	Actividad	Horas	Entregable
<b>FE-01</b>	Diseñar layout general	4	Plantilla base del frontend
<b>FE-02</b>	Panel de control	8	Vista consolidada del POA
<b>FE-03</b>	Formulario de registro de acciones	4	Creación de nuevas actividades
<b>FE-04</b>	Tabla de seguimiento	8	Listado interactivo de actividades
<b>FE-05</b>	Conectar vistas con APIs REST	8	Integración con backend

*Nota: Para este efecto se diseña en el framework de React*

Para este apartado se puede evidenciar que el desarrollo de la aplicación y los formularios que utilizaron para la interfaz del usuario.

### 3.10 SPRINT 5 – MEDIOS DE VERIFICACIÓN Y VALIDACIONES

Este sprint aborda la subida, validación y visualización de medios de verificación, elementos clave para garantizar el cumplimiento normativo del POA

Tabla 15

Sprint 5, medios de verificación y validaciones

ID	Actividad	Horas	Entregable
VER-01	Diseñar modelo de verificación	3	Estructura de documentos
VER-02	Subida de archivos adjuntos	5	Soporte para PDF, imágenes, word
VER-03	Validación obligatoria de evidencia	4	No se marca como completado sin medio
VER-04	Visualización desde el frontend	4	Descarga y previsualización
VER-05	Validar fechas límite	4	Alertas de vencimiento

Nota: Se detalla las acciones para el las validaciones de la aplicación

En la tabla anterior se puede observar las acciones a realizar del proyecto para las validaciones y verificación de las notificaciones, según sea el caso solicitado.

### 3.11 SPRINT 6 – GENERACIÓN DE INFORMES AUTOMÁTICOS

Este sprint se enfoca en la generación automática de informes por departamento, estas ayudan en la elaboración de lo informes para la rendición de cuentas y el control de la información que requieren las entidades de regulación de la educación, todas parten de lo que plantea en el POA.

Tabla 16

Sprint 6, generación de informes

ID	Actividad	Horas	Entregable
REP-01	Diseñar estructura de informe	4	Campos requeridos por departamento
REP-02	Generar informes PDF/Excel	6	Librerías integradas y funcional
REP-03	Botón de exportar informe	4	Funcionalidad en React
REP-04	Programar reportes mensuales	6	Script de generación periódica
REP-05	Vincular informes con metas	4	Informe dinámico según avance

*Nota: Sprint dedicado a la revisión de como se vinculan con la maquina*

Podemos observar que la tabla anterior se denota las funcionalidades y acciones que se deben requerir.

### 3.12 SPRINT 7 – CI/CD Y AUTOMATIZACIÓN

Este sprint cubre la automatización del flujo de desarrollo, pruebas y despliegue, asegurando calidad continua y entrega confiable del sistema en producción a través de SonarCloud, descritos en el numeran anterior.

Tabla 17

Sprint 7, CI/CD

ID	Actividad	Horas	Entregable
<b>CI-01</b>	Configurar GitHub Actions (CI)	4	Pipeline de pruebas unitarias
<b>CI-02</b>	Configurar GitHub Actions (CD)	4	Despliegue automático en Ubuntu
<b>CI-03</b>	Integrar SonarCloud	4	Análisis automático de código
<b>CI-04</b>	Respaldos automáticos de BD	4	Script de backup en Ubuntu
<b>CI-05</b>	Validar pipeline completo	4	Flujo de CI/CD probado

*Nota: Se detalla los aspectos que se realizaran en la plataforma de GitHub*

En esta etapa se crean los pipelines que permite realizar el CI/CD y que una vez que vaya a producción se alojara en el servidor local del instituto.

### 3.13 SPRINT 8 – DOCUMENTACIÓN Y PRUEBAS FINALES

Para este último sprint se redacta la documentación técnica y la de usuario, además de realizar pruebas finales en entorno local antes del lanzamiento.

Tabla 18

*Sprint 8, Documentación y pruebas finales*

ID	Actividad	Horas	Entregable
<b>DOC-01</b>	Redactar manual de usuario	10	Guía para jefes de área
<b>DOC-02</b>	Redactar manual técnico	8	Arquitectura y guía de instalación
<b>DOC-03</b>	Realizar casos de prueba	6	Escenarios validados
<b>DOC-04</b>	Revisión final del sistema	6	Sistema estable y probado
<b>DOC-05</b>	Despliegue en servidor local	6	Versión funcional en Ubuntu

*Nota: En este apartado se detalla*

Para el acceso de la aplicación se lo realizará a través del servidor local

### 3.14 VALIDACIÓN FUNCIONAL

Para garantizar que la aplicación cumple con lo establecido en este proyecto se considerará lo mencionado en la tabla siguiente:

Tabla 19

Criterios de verificación y validación de la aplicación

Criterios	Aspecto a evaluar
<p><b>Verificación de las acciones registradas en el POA</b></p>	<ul style="list-style-type: none"> <li>• La aplicación en su interfaz permite la subida del documento oficial del POA y se visualiza según su periodo</li> <li>• Registros de progreso de cada acción a realizar</li> <li>• Subir medios de verificación</li> <li>• Validación para el registro de cada acción</li> <li>• Mostrar notificaciones de vencimiento programadas</li> <li>• Permitir filtrar, buscar y exportar datos</li> </ul>
<p><b>Pruebas técnicas</b></p>	<ul style="list-style-type: none"> <li>• Pruebas unitarias, que validen el funcionamiento individual de cada componente del backend, frontend</li> <li>• Pruebas de integración, a través de la verificación de la comunicación entre los microservicios, bases de datos y el API Gateway</li> <li>• Pruebas de seguridad, se evalúa la autenticación OAuth2</li> </ul>
<p><b>Validación con usuarios</b></p>	<ul style="list-style-type: none"> <li>• Sesionar con las cuentas institucionales</li> <li>• Registrar acciones y metas</li> <li>• Actualizar estados de actividades</li> <li>• Subir medios de verificación</li> </ul>

Criterios	Aspecto a evaluar
	<ul style="list-style-type: none"><li>Mostrar informes gráficos</li></ul>

*Nota: En la tabla podremos observar las acciones a realizar para la validación del proyecto*

Al momento de realizar pruebas técnicas y la validación con los usuarios se busca evidenciar que la aplicación haya tenido una adopción efectiva.

## 4. RESULTADOS Y DISCUSIÓN

### 4.1 DESCRIPCIÓN GENERAL DE LA SOLUCIÓN DESARROLLADA

La aplicación web denominada POAitec está alineada con que se pueda centralizar u optimizar el proceso del seguimiento y control del Plan Operativo Anual; para que de esta forma cada área a la cual se le asigne sus tareas y actividades, las mismas puedan cumplir con los estándares normativos ante la unidad de aseguramiento de la calidad.

Mediante esta aplicación se busca resolver problemáticas como la vinculación entre las actividades y metas del PEDI, tales como:

- Identificar de manera rápida y clara sobre los responsables y avances de cada actividad
- Acceder de una mejor manera a los medios de verificación
- Evitar la fragmentación de archivos y no retrasen la toma de decisiones.

La aplicación está dirigida a los siguientes actores del AITEC:

- Administrador del sistema, gestionan usuarios, roles y configuraciones generales del sistema
- Unidad de aseguramiento de la calidad, supervisan el avance general del POA
- Responsables de actividades, es el personal a cargo de ejecutar y actualizar el estado de cada acción

El frontend está desarrollado en React JS, donde los usuarios podrán gestionar sus actividades visualizar el dashboards y dar un seguimiento a las acciones realizadas, para el backend se utilizó el framework Feathers JS, el proyecto tiene un enfoque de arquitectura de microservicios, la base de datos, para la facilidad de respaldo, comparten el mismo motor, pero las tablas están lógicamente separadas.

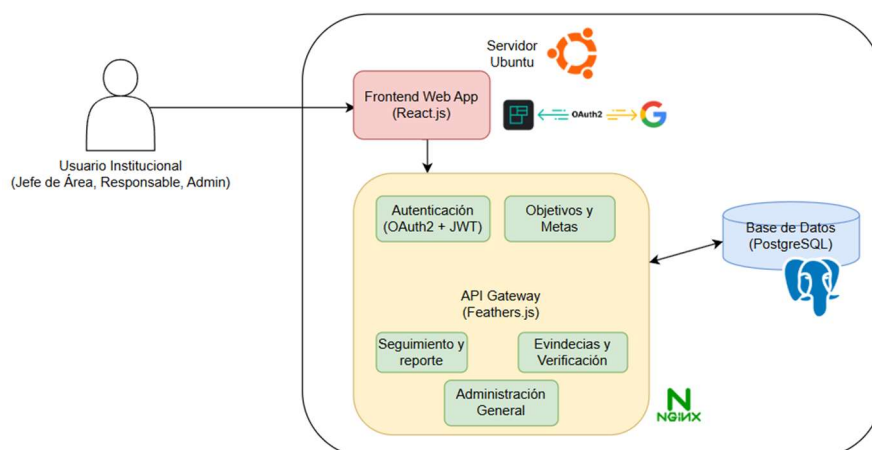
React permite con mucha facilidad que una página sea responsive, la aplicación al ser de tipo web, se puede aperturar en cualquier plataforma del sistema operativo y/o dispositivo, aunque se sesionan desde los PC's de escritorio de cada responsable inscrito en la aplicación.

Además, es compatible con los navegadores más utilizados en el medio, sin la necesidad de instalaciones adicionales.

El proyecto tiene como servidor Ubuntu, y tiene instalado Nginx para el balanceo de carga y seguridad del sitio.

*Ilustración 5.*

*Arquitectura macro de POAitec*



*Nota: En la imagen se puede observar la estructura general del POAitec*

De la imagen anterior tenemos la siguiente estructura tecnológica

Tabla 20.

Estructura Tecnológica.

Capa	Tecnología	Descripción
<b>Frontend</b>	React.js	Interfaz web dinámica y responsiva
<b>Backend</b>	Feathers.js + Node.js	Microservicios RESTful modularizados
<b>Base de Datos</b>	PostgreSQL	Almacenamiento seguro y centralizado
<b>Autenticación</b>	OAuth2 + Google Identity Platform	Inicio de sesión con credenciales institucionales
<b>Infraestructura</b>	Docker + Ubuntu Server	Contenedores aislados y servidor local
<b>Servidor Web</b>	Nginx	Proxy inverso y balanceador de carga
<b>Control de Versiones</b>	GitHub + GitHub Actions	CI/CD automatizado
<b>Análisis de Calidad</b>	SonarCloud	Detección de errores, vulnerabilidades y malas prácticas

*Nota: Se muestra las tecnologías utilizadas en el proyecto*

Este sistema mejora la eficiencia operativa; además de fortalecer el cumplimiento normativo y la transparencia en sus procesos, lo cuales permiten mantener la acreditación institucional.

## 4.2 MÓDULOS Y FUNCIONALIDADES IMPLEMENTADAS

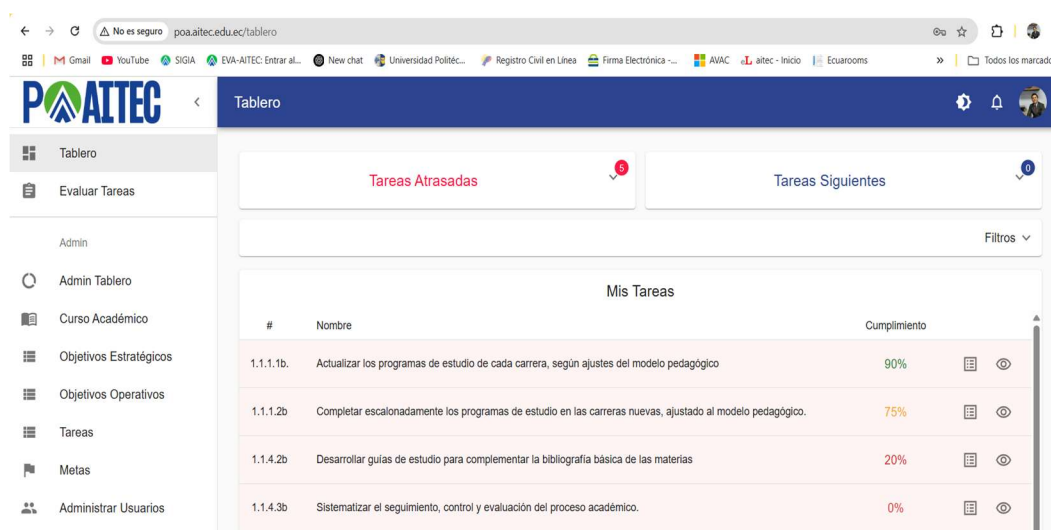
Las funcionalidades claves que se tiene la aplicación web son:

- **Tablero**, en este apartado se podrán observar las tareas asignadas; así como también, un control de tareas atrasadas y cuales son las siguientes, según sea las actividades asignadas
- **Evaluar tareas**, se visualizan las tareas asignada, medios de verificación, responsables de dicha actividad
- **Administrar Tablero**, grafico donde se visualizan las actividades y tareas asignadas

- **Curso académico**, se registra el periodo lectivo y se sube el documento oficial del POA en un periodo determinado
- **Objetivos estratégicos**, se registran los objetivos estratégicos
- **Objetivos operativos**, se registran los objetivos operativos de cada objetivo estratégico
- **Tareas**, se registran las tareas por cada objetivo operativo, quien es el verificador o supervisor, los responsables, periodo de asignación de cumplimiento de la tarea, las metas, y un presupuesto según sean el caso
- **Metas**, se registran las metas de cada curso académico
- **Administración de usuarios**, se registran los usuarios por su rol:
  - Administrador, el que gestiona a los usuarios y la aplicación
  - Supervisor cuyos actores son los integrantes de la Unidad de Aseguramiento de la Calidad, también puede existir algún otro usuario en especial que se le asigne la labor de verificar las metas asignadas, y
  - Usuario quienes son actores responsables por cada área en particular de la institución

*Ilustración 6.*

*Entorno del POAitec con el rol de administrador*



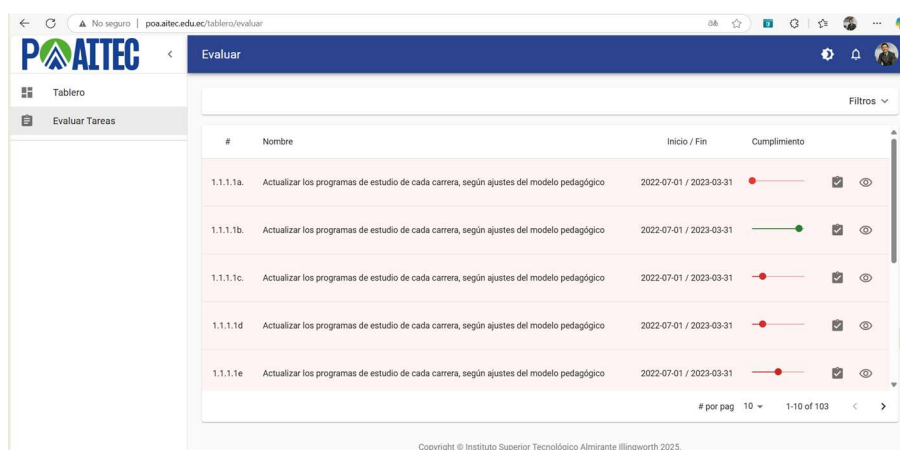
#	Nombre	Cumplimiento
1.1.1.1b.	Actualizar los programas de estudio de cada carrera, según ajustes del modelo pedagógico	90%
1.1.1.2b	Completar escalonadamente los programas de estudio en las carreras nuevas, ajustado al modelo pedagógico.	75%
1.1.4.2b	Desarrollar guías de estudio para complementar la bibliografía básica de las materias	20%
1.1.4.3b	Sistematizar el seguimiento, control y evaluación del proceso académico.	0%

*Nota: En la imagen podemos observar el entorno principal de la aplicación*

En la imagen anterior podemos observar un entorno del rol de administrador, sin embargo, cuando es supervisor solamente se aplicación las funcionalidades de tablero y evaluar las tareas.

Ilustración 7.

Entorno del rol de supervisor del POAitec



*Nota: En esta sección el supervisor a cargo de la revisión de los usuarios y sus metas asignadas*

El rol de supervisor le da la oportunidad de revisar las asignaciones y los medio de verificación que hayan cargado los usuarios

Ilustración 8.

Entorno de la aplicación con el rol de Usuario del POAitec



*Nota: En esta sección se podrán cargar las evidencias y su avance según las tareas asignadas*

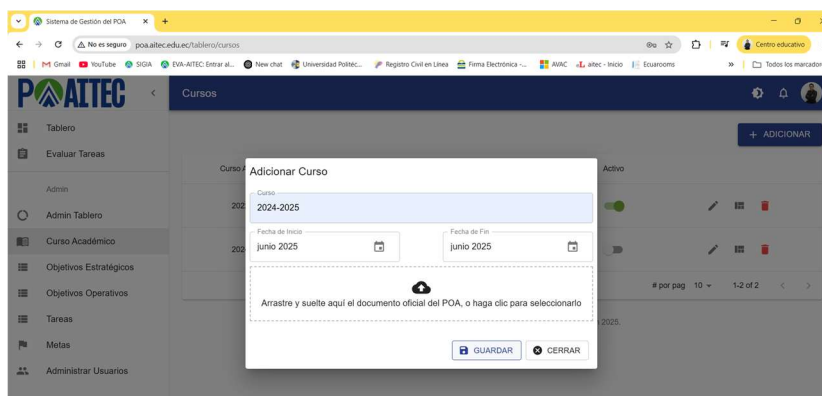
De forma adicional en la imagen podemos notar que se notifica al usuario que actividades tiene atrasada y cuáles serían las sigüientes a cumplir.

### 4.3 VALIDACIÓN FUNCIONAL DEL SISTEMA

Con la finalidad de asegurar que todas las funcionalidades desarrollados están cumpliendo con los requerimientos establecidos y las expectativas de los usuarios finales, en primera instancia debemos crear el curso del periodo académico correspondiente del POA como lo podemos observar en la ilustración siguiente:

Ilustración 9

Creación del curso correspondiente al periodo a evaluar

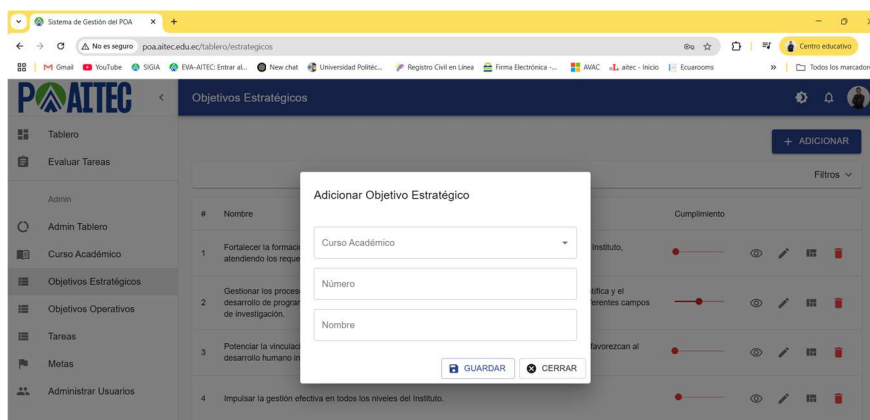


Nota: Imagen correspondiente a la creación del curso el cual debe tener el rol de administrador

A través de la creación del curso ya se puede adicionar los objetivos estratégicos para el POA.

Ilustración 10

Adicionar objetivos estratégicos del POA

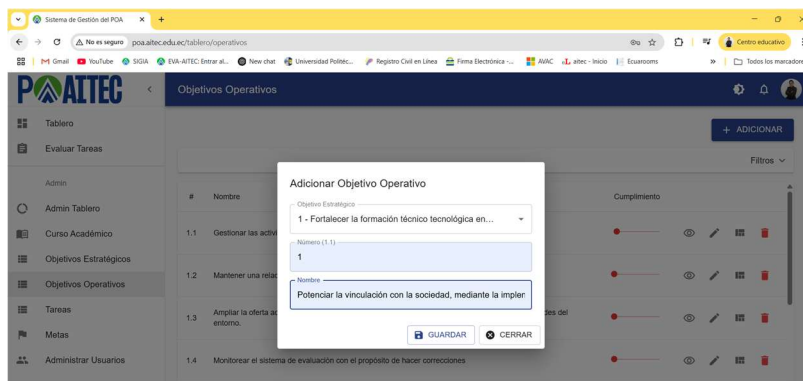


Nota: La finalidad de realizar tal aspecto es con la finalidad de poder agregar los objetivos operativos que tendrán cada una de las estrategias establecidas en el POA

Al momento que se agregan dichos objetivos, podemos establecer los objetivos operativos estas acciones tienen la finalidad de ir organizando la información de tal manera que posteriormente podamos indicar cuales serian las tareas asignadas para cada responsable.

Ilustración 11

Agregar objetivos operativos

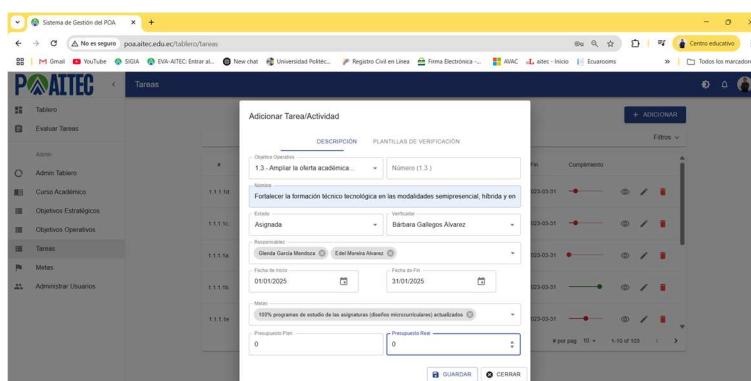


*Nota: Para poder agregar un objetivo operativo es indispensable agregar los objetivos estratégicos*

Con lo anterior expuesto y considerando que se agregaron las metas que tienen las distintas actividades se crean las tareas a realizar en el POA.

Ilustración 12

Agregar tareas para el POA



*Nota: Podemos observar que en este espacio se establecen quienes son los responsables y quien supervisará dichas actividades.*

Una vez que se determinaron y crearon todos los aspectos necesarios para poder asignar las tareas para los responsables de la información cuáles son las acciones a

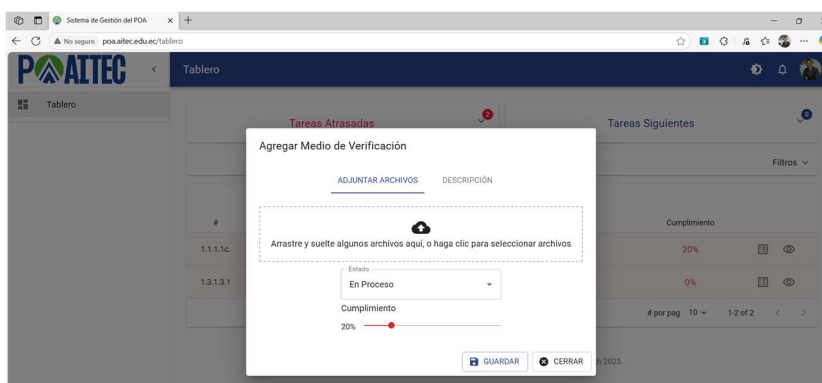
realizar para la asignación de una tarea, para validar cada una de los módulos, se aplicaron las siguientes estrategias:

- **Test funcional**, se validaron que los módulos expuestos respondieran de forma correcta y se den las acciones esperadas, entradas y flujos del sistema, tal como:
  - Registros de cursos, objetivos y otros
  - Asignaciones de tareas y responsables
  - Carga de evidencias
  - Actualización de estados
  - Visualización del tablero y su grafica
- **Pruebas con usuarios reales**, estas sesiones permitían que los actores que intervienen en la aplicación puedan interactuar con el sistema, realizando tareas como:
  - Revisar la facilidad de navegación
  - Claridad en las opciones propuestas
  - Respuesta visual de las acciones por cada tipo de usuario

En la imagen siguiente se puede observar como se visualiza las acciones y/o tareas que tiene un usuario especifico.

*Ilustración 13*

*Agregar evidencias del usuario*



*Nota: Esta ventana tiene la posibilidad de subir las evidencias y redactar la pestaña descripción los aspectos realizadas para la tarea*

En bases a las observaciones recibidas se realizaron ajustes a nivel de interfaz, tales como:

- Mejora en los filtros de búsqueda
- Validación de adicional para cargas incompletas
- Que las acciones sean más intuitivas en su navegación

A nivel técnico se revisaron

- Seguridad en el inicio de sesión de Google OAuth2
- Desempeño del sistema
- Compatibilidad con navegadores
- RespalDOS de las bases de datos

Con todo lo expuesto en la tabla siguiente podemos revisar lo acontecido en cada aspecto validado

*Tabla 21*

*Validaciones del sistema*

<b>Funcionalidad</b>	<b>Resultado de la Prueba</b>	<b>Observaciones</b>
<b>Inicio de Sesión con Google</b>	Funciona correctamente	Acceso rápido y seguro con credenciales institucionales
<b>Registro de Objetivos Estratégicos</b>	Funciona correctamente	Intuitivo y fácil de realizar
<b>Registro de Objetivos Operativos</b>	Funciona correctamente	Existe una relación entre objetivos estratégicos
<b>Asignación de Tareas</b>	Funciona correctamente	Se pueden asignar múltiples responsables y fechas límite, cual es el supervisor, sus metas y el estado de la

<b>Funcionalidad</b>	<b>Resultado de la Prueba</b>	<b>Observaciones</b>
		misma, sin embargo, debe ser más intuitivo
<b>Carga de Medios de Verificación</b>	Funciona correctamente	Se aceptan PDF's, imágenes y documentos Word
<b>Actualización del Estado de Actividades</b>	Funciona correctamente	Estados actualizables
<b>Generación de Informes PDF/Excel</b>	Funciona correctamente	Se puede visualizar correctamente la información subida por los responsables de cada tarea sin ninguna novedad
<b>Panel de Control General</b>	Funciona correctamente	Muestra claramente el avance del POA por departamento
<b>Administración de Usuarios</b>	Funciona correctamente	Roles bien definidos y fáciles de gestionar
<b>Configuración de Periodos Académicos</b>	Funciona correctamente	Posibilidad de registrar periodos y asociar actividades y cargar el POA

*Nota: Podemos encontrar las funciones del proyecto*

En base a lo observaciones recibidas por parte del personal del AITEC, se realizaron los siguientes ajustes:

- Validación para evitar cargas incompletas
- Mejorar los filtros de búsquedas
- Rediseño del formulario de registro de metas para que sea más intuitivo

Con base a las validaciones realizadas, el sistema cumple con los objetivos planteados para las necesidades solicitadas en las problemáticas planteadas.

## 4.4 EVALUACIÓN DE CUMPLIMIENTO DE OBJETIVOS

De manera general el proyecto cumplió satisfactoriamente con el objetivo para esta propuesta; sin embargo, las funcionalidades esta sujetas a mejora visuales y de rendimiento; como objetivo general planteado en esta propuesta podemos indicar lo siguiente:

- La aplicación fue diseñada con arquitectura de microservicios y desarrollado con metodologías ágiles
- Con React JS. Se tiene un frontend responsivo
- Se aplicaron buenas prácticas de CI/DC
- El POA se volvió una herramienta digital y accesible en cualquier dispositivo

Para analizar el cumplimiento de los objetivos específicos observaremos la siguiente tabla.

Tabla 22

*Cumplimiento de objetivos*

Objetivo Específico	Funcionalidad Desarrollada	Nivel de Cumplimiento
<b>Estimar de manera precisa los requisitos funcionales y técnicos de la aplicación, asegurando una comprensión clara de las necesidades del Instituto.</b>	<ul style="list-style-type: none"> <li>• Se realizó un análisis de los problemas derivados del documento del POA</li> <li>• Se establecieron los requerimientos funcionales de la aplicación</li> <li>• Se recopiló información a través de reuniones, con la unidad de aseguramiento y las áreas responsables</li> </ul>	Cumplido
<b>Implementar iterativamente un</b>	<ul style="list-style-type: none"> <li>• Desarrollo incremental en Sprints ágiles.</li> </ul>	Se cumplió con el objetivo

Objetivo Específico	Funcionalidad Desarrollada	Nivel de Cumplimiento
<b>prototipo funcional de la aplicación, permitiendo la retroalimentación constante para validar conceptos.</b>	<ul style="list-style-type: none"> <li>• Se realizaron validaciones con usuarios</li> <li>• Se realizaron ajustes sobre los módulos de tareas, metas y tablero.</li> </ul>	propuesto; sin embargo, requiere iteración final de reportes y notificaciones automáticas a los usuarios
<b>Adoptar prácticas ágiles de desarrollo de software durante todo el ciclo del proyecto, garantizando la flexibilidad para adaptarse a cambios.</b>	<ul style="list-style-type: none"> <li>• Uso de metodología Scrum,</li> <li>• Se realizaron reuniones y revisiones de los sprints</li> <li>• Se utilizo un control de versiones y automatización don Github Actions</li> <li>• Se monitoreo con SonarCloud, vulnerabilidades y errores de código</li> </ul>	Cumplido

*Nota: La tabla fue elaborada en base a todos los aspectos abordado en este proyecto*

Este proyecto mantuvo un enfoque ágil, definiendo iteraciones efectivas para el cumplimiento de la digitalización del POA como se observa la tabla anterior, todas las funcionalidades fueron probadas y cumplieron con las expectativas de la institución y los actores que lo requieren; sin embargo, en una nueva instancia se requiere que se mejoren los resultados de los informes orientados a una experiencia de usuario UX/UI y por ultimo las notificaciones a usuarios sobre las acciones realizadas y cumplidas debe ser más automatizadas.

## 5. CONCLUSIONES

---

El desarrollo de la aplicación web POAitec ha demostrado ser una solución viable y necesaria para optimizar la gestión del Plan Operativo Anual, la implementación de una plataforma digital centralizada, desarrollada bajo una arquitectura de microservicios y siguiendo buenas prácticas de desarrollo ágil y CI/CD, ha permitido superar las limitaciones del sistema manual previo, mejorando significativamente la organización, seguimiento y rendición de cuentas de las actividades institucionales.

Consecuentemente, el desarrollo de esta herramienta, permitió que los responsables de cada área pueden registrar, actualizar y monitorear sus tareas en tiempo real, cargar medios de verificación, poder revisar dentro de la aplicación alertas de vencimientos y observar las tareas siguientes a desarrollar, darle un seguimiento a las actividades asignadas, todo ello con acceso seguro mediante credenciales institucionales; por lo tanto; no solo mejora la transparencia interna, sino que también facilita la evaluación de los criterios que son examinados por parte de organismos reguladores como el CACES.

El proyecto cumple satisfactoriamente los objetivos establecidos: se estimaron con precisión los requisitos funcionales y técnicos, se desarrolló un prototipo funcional mediante sprints iterativos y se adoptaron prácticas ágiles durante todo el ciclo de desarrollo. Si bien se identifican oportunidades de mejora, especialmente en la experiencia de usuario (UX/UI) y automatización avanzada de notificaciones, la aplicación representa una mejora enorme en el manejo de los procesos administrativos y académicos del AITEC y su digitalización contribuye así a la sostenibilidad institucional y reconocimiento académico.

## REFERENCIAS

- Adochiei, F., Ciucu, R., Adochiei, I.-R., Argatu, F., Seritan, G., Enache, B., . . . Argatu, V. (2019). Innovative Devops for Artificial Intelligence. *Sciendo*, 19, 58-63. doi:10.1515/sbeef-2019-0011
- Alshuqayran, N., Ali, N., & Evans, R. (2016). A Systematic Mapping Study in Microservice Architecture. *2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)* (págs. 44-51). Macau, China: IEEE.
- Aristizabal, A., Mogollón Alarcón, D., & Rubiano Cotrino, S. (2024). Desarrollo de una plataforma integral para la gestión de SG-SST y análisis de riesgo. *Universidad del Bosque*. Bogotá. Obtenido de <https://repositorio.unbosque.edu.co/server/api/core/bitstreams/2f3b3e77-612c-4f96-9d47-d53fd32104e3/content>
- Asamblea Nacional República del Ecuador. (agosto de 2018). *Ley Orgánica Reformatoria a Ley Orgánica de Educación Superior*. Obtenido de Registro Oficial 297: [http://esacc.corteconstitucional.gob.ec/storage/api/v1/10\\_DWL\\_FL/eyJYXWZXRhIjoicm8iLCJ1dWlkIjoieMmNkMWNhYzEtNzI1ZC00ZWVmLTk4YTA4YTg3NTE5ZDE4MTk4LnBkZiJ9](http://esacc.corteconstitucional.gob.ec/storage/api/v1/10_DWL_FL/eyJYXWZXRhIjoicm8iLCJ1dWlkIjoieMmNkMWNhYzEtNzI1ZC00ZWVmLTk4YTA4YTg3NTE5ZDE4MTk4LnBkZiJ9)
- Barcia Cedeño, E., Tambaco Quintero, A., Angulo Quiñónez, O., Prado Zamora, M., & Valverde Prado, N. (2024). Análisis de tendencias y futuro de la Inteligencia Artificial en la Educación Superior: perspectivas y desafíos. *Ciencia Latina Revista Científica Multidisciplinar*, 8, 3061-3076. doi:[https://doi.org/10.37811/cl\\_rcm.v8i1.9637](https://doi.org/10.37811/cl_rcm.v8i1.9637)
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., . . . Thomas, D. (2001). *Manifiesto for Agile Software Development*. Obtenido de Agilemanifesto: <https://agilemanifesto.org/iso/es/principles.html>
- Bonilla, D., Ríos, M., & Álvarez, J. (2018). Gestión del personal como efecto de las capacitaciones y su impacto en el cumplimiento del plan operativo anual institucional: caso Universidad Técnica de Tungurahua. *Universidad Ciencia y Tecnología*. Obtenido de [https://www.academia.edu/download/61740679/ARTICULO\\_UTA\\_Y\\_UDA20200110-6963-19iczvt.pdf](https://www.academia.edu/download/61740679/ARTICULO_UTA_Y_UDA20200110-6963-19iczvt.pdf)
- Bryson, J. (2011). *Strategic planning for public and nonprofit organizations* (3rd ed.). San Francisco: Jossey-Bass Inc Pub.
- Canosa Ferreira, A. (2024). *SCRUM: teoría e Implementación práctica* (1 ed.). Madrid: RA-MA. Obtenido de <https://elibro.net/es/ereader/aitec/267660?page=174>
- Cepuc, A., Botez, R., Craciun, O., Ivanciu, I.-A., & Dobrota, V. (2020). Implementation of a Continuous Integration and Deployment Pipeline for Containerized Applications in Amazon Web Services Using Jenkins, Ansible and Kubernetes. *2020 19th Conference: Networking in Education and Research RoEduNet* (págs. 1-6). Bucharest, Romania: RoEduNet. doi:<https://doi.org/10.1109/RoEduNet51892.2020.9324857>
- Consejo de Aseguramiento de la Calidad de la Educación Superior. (mayo de 2021). *Modelo de evaluación externa 2024 con fines de acreditación para los Institutos*

- Superiores Técnicos y Tecnológicos*. Obtenido de CACES:  
[https://www.caces.gob.ec/wp-content/uploads/Documents/2022/DOCUMENTOS%20WEB%20INSTITUTOS/Modelo%202024%20aprobado%20por%20el%20pleno%20\(1\).pdf](https://www.caces.gob.ec/wp-content/uploads/Documents/2022/DOCUMENTOS%20WEB%20INSTITUTOS/Modelo%202024%20aprobado%20por%20el%20pleno%20(1).pdf)
- Consejo de Aseguramiento de la Calidad en la Educación Superior. (noviembre de 2021). *Plan estratégico institucional 2022-2025*. Obtenido de CES:  
<https://www.caces.gob.ec/wp-content/uploads/2022/02/PLAN-ESTRATEGICO-INSTITUCIONAL-2022-2025-VF-validado-1.pdf>
- Consejo de Evaluación, Acreditación y Aseguramiento de la Calidad de la Educación Superior. (2016). Informe final de la evaluación del entorno de aprendizaje de los institutos superiores técnicos y tecnológicos. *CEACES*, Pg. 14. Quito. Obtenido de  
[https://www.caces.gob.ec/documents/20116/236718/52/5281.afsh/5281\\_1.0.afsh](https://www.caces.gob.ec/documents/20116/236718/52/5281.afsh/5281_1.0.afsh)
- Correa Delgado, R. (15 de octubre de 2010). *Decreto Ejecutivo 517*. Obtenido de Corte Constitucional: <https://www.educacionsuperior.gob.ec/wp-content/uploads/downloads/2018/01/Decreto-Ejecutivo-No.-517-sobre-la-Fusion-de-SENACYT-a-la-Secretaria-Nacional-de-Educacion-Superior-Ciencia-Tecnologia-e-Innovacion.pdf>
- Di Meglio, S., & Lucio Starace, L. (2024). Evaluating Performance and Resource Consumption of REST Frameworks and Execution Environments: Insights and Guidelines for Developers and Companies. *IEEE Access*, *12*, 161649-161669. doi:10.1109
- Díaz, O., Muñoz, M., & Mejía, J. (2019). Responsive infrastructure with cybersecurity for automated high availability DevSecOps processes. *2019 8th International Conference On Software Process Improvement (CIMPS)* (págs. 1-9). Leon, Guanajuato, Mexico: IEEE. doi:10.1109/CIMPS49236.2019.9082439
- Enríquez Herrera, J. V. (agosto de 2019). Business Intelligence en los procesos de seguimiento y evaluación del plan operativo anual de la UPEC. *Repositorio Digital Universidad Uniandes*. Obtenido de  
<https://dspace.uniandes.edu.ec/handle/123456789/10212>
- Fritsch, J., Bogner, J., Zimmermann, A., & Wagner, S. (2019). From Monolith to Microservices: A Classification of Refactoring Approaches. En *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment* (Vol. 11350, págs. 128-141). doi:10.1007/978-3-030-06019-0\_10
- Galarza López, J., & Almuñías Rivero, J. (2015). La gestión de los riesgos de planificación estratégica en las instituciones de educación superior. *Revista Cubana de Educación Superior*, *34*(2), 45-53. Obtenido de  
[http://scielo.sld.cu/scielo.php?pid=S0257-43142015000200005&script=sci\\_arttext&tlng=en](http://scielo.sld.cu/scielo.php?pid=S0257-43142015000200005&script=sci_arttext&tlng=en)
- Guaman, D., Yaguachi, L., Cueva, S., Jaramillo, D., & Soto, F. (2018). Performance evaluation in the migration process from a monolithic application to microservices. *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)* (págs. 1-8). Cáceres, Spain: IEEE. doi:10.23919/CISTI.2018.8399148

- Hoda, R., Noble, J., & Marshall, S. (2012). Documentation strategies on agile software development projects. *International Journal of Agile and Extreme Software Development*, 1, 23. doi:10.1504/IJAESD.2012.048308
- Hron, M., & Obwegeser, N. (2022). Why and how is Scrum being adapted in practice: A systematic review. *Journal of Systems and Software*, 183, 111110. doi:10.1016/j.jss.2021.111110
- Instituto Superior Almirante Illingworth. (2023). *Modelo de vinculación con la sociedad*. Obtenido de Vinculaitec: <https://vinculacion.aitec.edu.ec/wp-content/uploads/2023/04/MODELO-VINCULACION-CON-LA-SOCIEDAD-2023.pdf>
- Jaramillo Acevedo, C. A., Gómez, J. P., & Ríos Patiño, I. (2017). Methodology to transform a monolithic software into a microservice architecture. *2017 6th International Conference on Software Process Improvement (CIMPS)* (págs. 1-6). Zacatecas, Mexico: IEEE. doi:10.1109/CIMPS.2017.8169955
- Jimenez Villar, J. (2016). *Descubre ReactJS*. Lean Publishing.
- Kumar, P., Parikh, A., Gandhi, P., & Sisodia, J. (2022). Monolithic to Microservices Architecture - A Framework for Design and Implementation. *2022 International Conference on Computer, Power and Communications ICCPC*, (págs. 90-96). Chennai, India. doi:10.1109/ICCPC55978.2022.10072238
- Kunda, D., Chihana, S., & Muwanei, S. (2017). Web Server Performance of Apache and Nginx: A Systematic Literature Review. *Computer Engineering and Intelligent Systems*, 8(2), 43-52. Obtenido de <https://www.researchgate.net/publication/329118749>
- Martinez Corona, J., Aragón Paulín, R., & Gil Nuño, B. (2015). Modelo de planeación estratégica: diseño para una institución de educación superior. *TECTZAPIC*, N. 1. Obtenido de <http://www.eumed.net/rev/tectzapic/2015/01/planeacion-estrategica.html>
- Martínez Rizo, F. (2010). Los indicadores como herramientas para la evaluación de la calidad de los sistemas educativos. *Sinéctica* (35), 1-17. Obtenido de [https://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S1665-109X2010000200004&lng=es&nrm=iso&tlng=es](https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-109X2010000200004&lng=es&nrm=iso&tlng=es)
- Martins, J. (septiembre de 2024). *Plan operativo: Definición, tipos y cómo elaborar uno para tu empresa en 5 pasos*. Obtenido de ASANA: <https://asana.com/es/resources/operational-planning>
- Mathews, D. (2025). Application Service Resilience In Cloud: An End-to-End Perspective. *Electronics theses and dissertations of Indian Institute of Science*. Obtenido de <https://etd.iisc.ac.in/handle/2005/6763>
- Mbuta, I. (2024). Software testing optimization and continuous integration with MS AZURE DevOps. *ResearchGate*. doi:10.13140/RG.2.2.24710.18247
- Menjívar Valencia, E. (2023). ¿Cómo enfrentarse a las tendencias globales de la Educación Superior? *Diá-logos*, 1, 5-8. doi:10.5377/dialogos.v1i26.16313
- Meta. (2023). *React Documentation*. Obtenido de React: <https://react.dev>
- Monte Galiano, J. (2016). *Implantar scrum con éxito* (1 ed.). Barcelona: UOC. Obtenido de <https://elibro.net/es/lc/aitec/titulos/58575>
- Moreno de la Cruz, C., Jesús Correia, L. F., & Vargas-Lombardo, M. (2023). Un enfoque de arquitectura de microservicio para mHealth orientado al monitoreo ambulatorio. *2023 18th Iberian Conference on Information Systems and*

- Technologies (CISTI)* (págs. 1-7). Aveiro, Portugal: IEEE.  
doi:10.23919/CISTI58278.2023.10211405
- Murillo Valverde, R. (2024). El Plan Estratégico de Desarrollo Institucional y el Plan Operativo Anual cómo instrumentos de gestión en las Instituciones de Educación Superior. *Revista InveCom*, 4(2). doi:10.5281/ZENODO.10632863
- Negrete Rodríguez, M. A., & Infante Martínez, F. U. (2022). Creation of serverless applications in the cloud. *2022 11th International Conference On Software Process Improvement (CIMPS)* (págs. 216-218). Acapulco, Guerrero, Mexico: IEEE. doi:10.1109/CIMPS57786.2022.10035685
- Newman, S. (2019). *Monolith to microservices: evolutionary patterns to transform your monolith*. O'Reilly Media, Inc.
- Oviedo Rodríguez, M., Medina León, A., Negrín Sosa, E., & Carpio Vera, D. (2017). El enfoque en procesos desde la planificación operativa en las Instituciones de Educación Superior del Ecuador. *Ingeniería Industrial*, 116-128. Obtenido de [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S1815-59362017000100011](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1815-59362017000100011)
- Pazmiño Moscoso, S. P. (diciembre de 2018). ANÁLISIS DEL SISTEMA DE EDUCACIÓN SUPERIOR EN ECUADOR DESDE EL 2000 HASTA LA ACTUALIDAD "OLAS, CIERRE Y CALIDAD". *Revista de la Escuela de Ciencias de la Educación*, 2(13), 53-68. Obtenido de [http://www.scielo.org.ar/scielo.php?script=sci\\_arttext&pid=S2362-33492018000200005&lng=es&tlng=es](http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S2362-33492018000200005&lng=es&tlng=es)
- Pinnington, A., & Mir, F. (2014). Exploring the value of project management: Linking Project Management Performance and Project Success. *International Journal of Project Management*, 32, 202-217. doi:10.1016/j.ijproman.2013.05.012
- Quelal, R., Villavicencio, M., & Mendoza, L. (2018). A survey of agile software development methodologies in Ecuador. *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)* (págs. 1-6). Caceres, Spain: IEEE. doi:10.23919/CISTI.2018.8399186
- Rising, L., & Janoff, N. (2000). The Scrum software development process for small teams. *IEEE Software*, 17, 26-32. doi:10.1109/52.854065
- Rojas Aquino, A., González Villagra, M., Flores Cardozo, M., Leiva de Céspedes, S., Cáceres Florentín, R., & Sánchez Sánchez, E. (2024). Control de gestión como parámetro de toma de decisiones en la Universidad Nacional de Caaguazú. *III Bienal Científica y Tecnológica Internacional - UNICAN 2024*. Saltos del Guairá, Paraguay.
- Saavedra, N., Silva, A., & Monperrus, M. (2024). GitBug-Actions: Building Reproducible Bug-Fix Benchmarks with GitHub Actions. *2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, (págs. 1-5). Lisbon, Portugal. doi:10.1145/3639478.3640023
- Salunke, S., & Ouda, A. (2024). A Performance Benchmark for the PostgreSQL and MySQL Databases. *Future Internet*, 16, 382. doi:10.3390/fi16100382
- Schwaber, K., & Sutherland, J. (Noviembre de 2020). *The Definitive Guide to Scrum: The Rules of the Game*. Obtenido de Scrum Guides: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- Silva, S., Palomino, P., Araújo, R., Araújo, A., & Dermeval, D. (2024). Mastering Requirements Volatility: Strategies for Dynamic Environments and Multiple

- Stakeholders in Government Software Projects. *IEEE Access*, 12, 183060-183077. doi:10.1109/ACCESS.2024.3510056
- Silva Rosas, J. E. (septiembre de 2019). *Elaboración de un Proceso Metodológico para mejorar la formulación, ejecución, seguimiento y evaluación del Plan Operativo Anual articulado al Plan Estratégico Institucional en una Gestión por resultados en la Municipalidad Provincial De Talara*. Obtenido de Universidad de Sipan: <http://repositorio.uss.edu.pe//handle/20.500.12802/5879>
- Team, F. B. (2024). Continuous Integration and Continuous Deployment. En F. B. Team, *Cloud-Native Application Architecture: Microservice Development Best Practice* (págs. 351-382). Springer, Singapore. doi:10.1007/978-981-19-9782-2\_9
- Veeranjaneyulu Veeri. (2024). Performance Optimization Techniques in React Applications: A Comprehensive Analysis. *International Journal of Research in Computer Applications and Information Technology*, 7(2), 1165–1177. doi:10.5281/ZENODO.14146734
- Villamar , M., Garcés , O., Castro, H., Verano, M., Salamanca, L., Casallas, R., & Gil, S. (2015). Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud. *2015 10th Computing Colombian Conference (10CCC)* (págs. 583-590). Bogota, Colombia: IEEE. doi:10.1109/ColumbianCC.2015.7333476
- Yu, J., Zhang, J., Sheng, Y., Mei, Y., Chen, Y., & Zhu, C. (2020). Design and Implementation of a Small Examination System Based on Scrum. *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)* (págs. 420-423). Beijing, China: IEEE. doi:10.1109/ICAICE51518.2020.00089