



ESCUELA DE POSGRADO
PROGRAMA ACADÉMICO MAESTRÍA EN INGENIERÍA DE
SISTEMAS CON MENCIÓN EN TECNOLOGÍAS DE LA
INFORMACIÓN

Modelo de Gestión de Proyectos para mejorar el Proceso de
Desarrollo de Software de la Corte Superior de Justicia de
Lambayeque, 2021

TESIS PARA OBTENER EL GRADO ACADÉMICO DE:

Maestra en Ingeniería de Sistemas con mención en Tecnologías de la
Información

AUTORA:

Carrasco Zeña, Jessica Katherin (ORCID: 0000-0003-2728-8588)

ASESOR:

Dr. Pacheco Torres, Juan Francisco (ORCID: 0000-0002-8674-3782)

LÍNEA DE INVESTIGACIÓN:

Sistemas de Información y Comunicaciones

TRUJILLO – PERÚ

2022

Dedicatoria

Dedico esta investigación a mis padres, y a mi hermana por su amor, su apoyo y brindarme sus modelos a seguir, que ha influenciado en mi vida dándome su consejo y guía para seguir satisfactoriamente mi formación profesional. Y por último a mi asesor de tesis que me ayudó en poder culminar esta investigación.

Jessica Katherin Carrasco Zeña

Agradecimiento

Con mucho agradecimiento a Dios, por ser mi guía en todo momento y ponerme en el camino personas que han aportado grandemente en mi carrera y mi formación como persona. Asimismo, quiero dedicar el presente proyecto de investigación a mis padres, porque ellos siempre estuvieron a mi lado brindándome su apoyo y consejo.

Jessica Katherin Carrasco Zeña

Índice de contenidos

Carátula	i
Dedicatoria	ii
Agradecimiento	iii
Índice de contenidos	iv
Índice de tablas	v
Índice de gráficos y figuras	vii
Resumen	viii
Abstract	ix
I. INTRODUCCIÓN	1
II. MARCO TEÓRICO	5
III. MÉTODO	15
3.1. Tipo y diseño de investigación	15
3.2. Variables y operacionalización	16
3.3. Población, muestra y muestreo	16
3.4. Técnicas e instrumentos de recolección de datos	17
3.5. Procedimientos.....	20
3.6. Método de análisis de datos.....	22
3.7. Aspectos éticos	25
IV. RESULTADOS	26
V. DISCUSIÓN	37
VI. CONCLUSIONES	43
VII. RECOMENDACIONES	44
REFERENCIAS	45
ANEXOS	

Índice de tablas

Tabla 1. Técnicas e instrumentos de investigación.	18
Tabla 2. Tabla Índice de Validez de Contenido (Lawshe).	18
Tabla 3. Valores de consistencia interna mediante el coeficiente Kuder- Richardson.....	19
Tabla 4. Valores de consistencia interna mediante el coeficiente McDonald's.....	20
Tabla 5. Hipótesis, el nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.....	22
Tabla 6. Hipótesis, el tiempo promedio de entrega de desarrollo de software	23
Tabla 7. Hipótesis, el nivel de comunicación con el área usuaria durante el desarrollo del software.....	23
Tabla 8. Hipótesis, el nivel de satisfacción del usuario al finalizar el desarrollo del software	24
Tabla 9. Estadísticos descriptivos del indicador nivel de conocimiento sobre metodologías ágiles.....	26
Tabla 10. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test del cuestionario de nivel de conocimiento sobre metodologías ágiles.	27
Tabla 11. Rangos obtenidos en el nivel de conocimiento sobre metodologías ágiles antes y después.	27
Tabla 12. Estadísticos de prueba	28
Tabla 13. Estadísticos descriptivos del indicador tiempo promedio para la entrega del proyecto de software.....	28
Tabla 14. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test de la ficha de registro de tiempo promedio para la entrega del proyecto de software.....	29
Tabla 15. Rangos obtenidos en el tiempo promedio para la entrega del proyecto de software antes y después	30
Tabla 16. Estadísticos de prueba	31
Tabla 17. Estadísticos descriptivos del indicador nivel de comunicación con el área usuaria.....	31

Tabla 18. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test de la ficha de registro de nivel de comunicación con el área usuaria	32
Tabla 19. Rangos obtenidos en el nivel de comunicación con el área usuaria antes y después	33
Tabla 20. Estadísticos de prueba	33
Tabla 21. Estadísticos descriptivos del indicador nivel de satisfacción de los usuarios al finalizar el desarrollo del software	34
Tabla 22. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test del cuestionario de nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia.....	35
Tabla 23. Rangos obtenidos en el nivel de satisfacción de los usuarios al finalizar el desarrollo del software antes y después.....	36
Tabla 24. Estadísticos de prueba	36

Índice de gráficos y figuras

Figura 1. Esquema de diseño pre experimental de investigación	15
Figura 2. Promedio del nivel de conocimiento sobre metodologías ágiles	26
Figura 3. Tiempo promedio para la entrega del proyecto de software.	29
Figura 4. Promedio del nivel de comunicación con el área usuaria.....	32
Figura 5. Promedio del nivel de satisfacción de los usuarios	34

Resumen

El presente estudio en curso tiene como propósito general Mejorar los Procesos de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque a través de la implementación de un Modelo de Gestión de Proyectos, además presenta como objetivos específicos: Determinar el nivel de conocimiento sobre las metodologías ágiles, reducir el tiempo promedio para la entrega del software, determinar el nivel de comunicación entre los involucrados, determinar el nivel de satisfacción de los usuarios. El estudio es de tipo aplicada, con enfoque cuantitativo y diseño experimental, contando con 25 colaboradores para la muestra, como instrumentos se aplicó dos cuestionarios y dos fichas de observación para comparar cada objetivo específico. Se hizo uso de la prueba de normalidad Shapiro-Wilk y para contrastar la hipótesis se utilizó la prueba Z de Wilcoxon, Chi-cuadrado para datos sin normalidad. Donde los resultados mostraron que se logró aumentar el promedio del nivel de conocimiento a un 17%, reducir el tiempo promedio a 9%, aumentar el nivel de comunicación a 95% y aumentar la satisfacción usuaria a 18%. Concluyendo que la implementación de un Modelo de Gestión de Proyectos mejoró los Procesos de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque.

Palabras clave: Proceso de desarrollo de software, Modelo de Gestión de Proyectos, Metodologías ágiles, Tiempo promedio, Comunicación entre involucrados.

Abstract

The general purpose of this ongoing study is to improve the Software Development Processes of the Superior Court of Justice of Lambayeque through the implementation of a Project Management Model. The specific objectives are: to determine the level of knowledge about agile methodologies, to reduce the average time for software delivery, to determine the level of communication among those involved, and to determine the level of user satisfaction. The study is of applied type, with quantitative approach and experimental design, counting with 25 collaborators for the sample, as instruments two questionnaires and two observation sheets were applied to compare each specific objective. The Shapiro-Wilk normality test was used and to contrast the hypothesis, the Wilcoxon Z test and Chi-square for non-normal data were used. The results showed that it was possible to increase the average level of knowledge to 17%, reduce the average time to 9%, increase the level of communication to 95% and increase user satisfaction to 18%. Concluding that the implementation of a Project Management Model improved the Software Development Processes of the Superior Court of Justice of Lambayeque.

Keywords: Software development process, Project Management Model, Agile Methodologies, Average time, Communication between stakeholders.

I. INTRODUCCIÓN

Pese a las buenas prácticas sobre la gerencia de proyectos en el mundo las fallas e incumplimiento de un proyecto de software se debe a varios factores económicos, tecnológico y geográfico. En el contexto internacional, Project-success-qrc-standish-group-chaos-report hace mención sobre el éxito de los proyectos de software, reportando que el 31 % de proyectos culmina con éxito, el 19% de proyectos fracasan y el 50% son reorientados, esto puede ser atribuido a la poca comunicación con el área usuaria y el equipo de desarrollo. (Project-success-qrc-standish-group-chaos-report, 2020)

Asimismo, la investigadora Alejandra Cuadros menciona que los proyectos de software fallan por muchos motivos y cita a Bob Lawhorn quien señala que en las empresas estadounidenses el fracaso de un proyecto representa una pérdida en dólares de por lo menos 30 mil millones cada año, asimismo, el 60% y el 80% de los fracasos en proyectos de TI de debe a insuficiente recopilación de requisitos y falta análisis de lo que se pretende lograr. (Cuadros, 2021)

En el ámbito nacional según lo mencionado en una publicación de la Universidad de ESAN menciona un estudio de Pulse of the Profession donde se detalla las más causas más resaltantes del fracaso de un proyecto entre ellas: el cambio de alcance del proyecto es del 37%, el 29% de empresas tiene una comunicación inadecuada o deficiente, el 25% no realiza una buena planificación de tiempos en los entregables, el 22% de ellas tienen un director inexperto en proyectos de software y el 13% de las empresas, no priorizan las tareas de una manera correcta con el equipo de desarrollo. (Universidad ESAN, 2019)

El desarrollo de software es clave en la digitalización de las empresas, es una actividad que ha ido evolucionando en las últimas décadas, con la llegada de nuevas herramientas tecnológicas que permita analizar, diseñar, probar e implementar software, por lo que todas las organizaciones tienen la necesidad de una constante mejora y de lograr una mejora constante y, en un corto período de tiempo, ciclos de desarrollo que correspondan a los objetivos propuestos. Sin embargo, a pesar del avance tecnológico existen algunas empresas que comparten algunos aspectos de la inestabilidad, donde el desarrollo de proyectos de software es un procedimiento de varios pasos extremadamente

complejo que puede complicarse si no se planifica, organiza y ejecuta cuidadosamente.

Actualmente, algunas empresas realizan proyectos de desarrollo de software mediante actividades manuales y / o métodos robustos, que pueden resultar engorrosos e ineficientes en muchos casos. Esta situación ha traído algunos problemas relacionados con la oportuna producción de software, ágil, de bajo costo y de alta calidad, es por ello que se recomienda usar métodos desde una perspectiva ágil, agregando el formalismo y abstracción requerida para las tareas más críticas logrando la automatización y optimización al proceso de desarrollo de software esto genera valor al negocio y mejora significativamente el proceso del software.

Por otro lado, es importante precisar que en las empresas del sector público lo que afecta el desempeño final de un proyecto de software es la ausencia de metodologías de desarrollo/ diseño no probadas falta de habilidades de gestión de proyectos, poca iniciativa para segmentar proyectos en tamaños más manejables y ausencia de herramientas que puedan ayudar a dar un seguimiento a los proyectos. (Barros, 2016)

Esta situación es lo que viene ocurriendo en un máximo órgano institucional del estado, la Corte Superior de Justicia de Lambayeque actualmente dicha institución ha ampliado sus competencias y fortalecido su infraestructura tecnológica, renovándose con la propuesta de una serie de proyectos de software que busca la modernidad en sus acciones, sin embargo, puede mostrar algunas limitaciones por no tener una buena organización, distribución y herramientas que ayuden con el rendimiento y la integración de todo el entorno de trabajo para lograr el éxito de un proyecto de software.

En Lambayeque La Corte Superior de Justicia, en conjunto con el Área de Informática y Sistemas está orientada al despliegue de nuevas tecnologías para ser efectivos los servicios públicos, pero también desea renovar sus procesos de desarrollo de aplicaciones, ya que todo este tiempo han tenido algunas limitaciones según el Ing. Marco Cespedes Burga Coordinador del área, el desarrollo de proyectos es iniciado al tomar las necesidades de la institución y todas sus sedes, la presidencia es quien autoriza el inicio de nuevos proyectos

internos, es en ese momento donde el área realiza la captura de requerimientos para el desarrollo del mismo, sin embargo, el problema sucede por la falta de implementación de una metodología que cumpla con los planes de trabajo o proyectos de desarrollo. Por otro lado, el equipo de desarrollo no cuenta con los conocimientos y recursos esenciales para la realización del proyecto, de manera que, fácilmente les permite incurrir en omisiones y errores en la organización tales como el incumplimiento del alcance, entregables y cronograma, incremento de recursos.

Entre los problemas principales del área es que no se cumple con la fecha planificada, debido a la falta de segmentación del proyecto, agregando algunos requerimientos que no están dentro del alcance del proyecto, por otro lado, se evidencia la falta de compromiso de los involucrados, en muchos casos se solicita el desarrollo de un proyecto, pero se necesita que todos formen parte de las reuniones desde el inicio hasta el cierre del mismo, sin embargo, no ha sabido manejar el tiempo de las coordinaciones y colaboradores no participan activamente debido a las múltiples actividades administrativas, el tiempo y la falta de compromiso ocasiona que el proyecto no marche de manera correcta.

Se puede decir que la gestión de un proyecto no es fácil, no solo se tiene terminar a tiempo, sino que además se debe realizar ajustándose al presupuesto y con la calidad suficiente para cumplir las expectativas del cliente, ante lo expuesto en la problemática la falta de planificación en tiempo, alcance y costo ocasiona la insatisfacción del área usuaria.

Es así como nos planteamos la interrogante, ¿De qué manera un modelo de Gestión de Proyectos influirá en los Procesos de Desarrollo de Software de la Corte Superior Justicia de Lambayeque, 2021?

Una vez implementado el modelo de gestión de proyectos sirven para crear valor y beneficio para la organización al dominar tareas, procesos y datos con el fin de armonizar los procesos dentro de la organización y cumplir con los cronogramas, la productividad y las estimaciones de calidad. Asimismo, a través de un marco de trabajo de gestión de proyectos permitirá implementar una plataforma informática en donde se permita elevar la productividad y la gestión de la calidad de sus procesos obteniendo equipos integrados especializados que trabajen de

forma conjunta de esta forma al aplicar un modelo de gestión de proyectos se incrementará la satisfacción de los interesados con respecto a los proyectos de software, cumpliendo con los requisitos establecidos.

El presente estudio en curso tiene como propósito general Mejorar los Procesos de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque a través de la implementación de un Modelo de Gestión de Proyectos, además presenta como objetivos específicos los cuales son: Determinar el nivel de conocimiento de los equipos de desarrollo sobre las metodologías ágiles de gestión de proyectos en la Corte Superior de Justicia. Reducir el tiempo promedio para la entrega del software con el apoyo de una plataforma informática que permita planificar los requerimientos de la Corte Superior de Justicia. Determinar el nivel de comunicación entre los involucrados del desarrollo del software de la Corte Superior de Justicia. Determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la Corte Superior de Justicia

Así mismo, se planteó la siguiente hipótesis alterna de esta investigación, un Modelo de Gestión de Proyectos mejorará significativamente los Procesos de Desarrollo de Software de la Corte Superior Justicia de Lambayeque.

A su vez, se planteó la siguiente hipótesis nula de esta investigación, un Modelo de Gestión de Proyectos no mejorará significativamente los Procesos de Desarrollo de Software de la Corte Superior Justicia de Lambayeque.

II. MARCO TEÓRICO

Para el estudio se tomaron en consideración los antecedentes de investigación presentados a continuación:

A nivel Internacional se consideraron las siguientes investigaciones:

En el artículo científico de Castro et al. (2020) que fue publicado en la revista Scielo, tuvo como objetivo principal plantear una mejora en el potencial de alcanzar el éxito sobre el proceso y acumulación de nuevos recursos tecnológicos. Como enfoque se empleó una investigación aplicada basada en el método evolutivo incremental con el objetivo de analizar, diseñar e implementar software para el incremento de nuevas funcionalidades en la satisfacción de necesidades requeridas. En los resultados se cerró el impedimento entre proyectos de desarrollo de software culminados con éxito y lo que no lo fueron, siendo los últimos en los que no se tiene en cuenta una gestión de riesgos. Los autores concluyen que fue de gran aporte porque permitió conocer el método evolutivo incremental, similar al SCRUM donde incluye diferentes etapas desde la especificación de requerimientos hasta la ejecución de test de prueba del software, lo que permite evidenciar requisitos cambiantes prolongados durante el tiempo del ciclo de vida del proceso de desarrollo de software.

En la investigación de Kyoo (2016) titulado “Desarrollo de software innovador y gestión de proyectos macro para empresas emergentes de tecnología” publicado por el Repositorio de la Universidad de La Salle. El principal objetivo fue mejorar las fases del desarrollo de software en adaptación con poco conocimiento de gestión de proyectos. La investigación de tipo aplicada y explicativa tomó como población a empresas emergentes de tecnología. En los resultados permitió conocer la adaptación real del proceso SIM con la gestión de un proyecto completo, reduciendo de manera efectiva la carga laboral de las empresas emergentes en tecnología. El autor concluye que un software basado en la innovación sistemática montada SIM, permite generar ideas innovadoras de manera sistemática.

En la investigación de Rodríguez (2016) que fue publicado por el Repositorio de la Universidad EAN en Bogotá, Colombia. El principal objetivo fue consolidar las prácticas y metodologías utilizando un marco de trabajo de gestión de proyectos

de TI funcional para el desarrollo de software. Como instrumento se empleó la encuesta y entrevista. En los resultados se conocieron e identificaron la importancia de una gestión de proyectos consolidando el uso de mejores prácticas y metodologías de las organizaciones. Tal es el caso, COBIT declara a la empresa el requerimiento de ejecutar diferentes proyectos para desarrollar software, PMBOK gestiona la realización de proyectos y CMMI-DEV se utiliza para el desarrollo de las actividades del proceso para desarrollar software. ITIL coloca el resultado de software como solución en el entorno de producción para garantizar un funcionamiento eficiente. El autor concluye que, las tecnologías desarrolladas en las organizaciones no siguen una metodología que gestione proyectos de desarrollo de software, obteniendo soluciones incompletas e inestables.

En la investigación de Maldonado & Cueva (2019) que fue publicado por el Repositorio de la Universidad Técnica Particular de Loja. El principal propósito fue mostrar las ventajas de ofrecer un marco de trabajo de gestión de proyectos para desarrollar software. En los resultados la investigación indica que no hay mejores o peores métodos para desarrollar un software y gestionar un proyecto que otros. Estos se pueden utilizar según el tipo de entorno en el que se implementen y el tipo de proyectos que desee gestionar. Se concluye que un modelo para gestionar de proyectos que se haga uso para controlar el proceso de desarrollo de software es importante, porque una metodología para gestionar proyectos de software eficaz se asocia con ventajas brindadas por el software orientado a modelos que brinda un modelo de calidad.

En la tesis de Rudas (2017) que fue publicado por el Cirteq, contó con el propósito general de diseñar e implementar propuesta de un modelo para gestionar riesgos prácticos y efectivos en proyectos. En los resultados se pudo observar que los efectos positivos de implementar un marco de trabajo de gestión de riesgos dentro de una organización centrada en el mercado de desarrollo de proyectos se traducen en ventajas palpables para reducir costos, asegurar la calidad y disminuir el tiempo de desarrollo del proyecto tal como se planeó. El estudio llega a la conclusión que al implementar un modelo de para gestionar riesgos en el proceso de la gestión de proyectos, los resultados

evidencian un beneficio actual y futuro, en donde cada negocio responde proactivamente a los riesgos y evite improvisaciones cuando ocurran eventos adversos, así se obtendrá un aumento de resultados positivos en los objetivos de costo y calidad.

A nivel Nacional se consideraron las siguientes investigaciones:

En la investigación de Illicachi y Olmedo (2020) que fue publicado por el repositorio de la Escuela de Posgrado Neumann. El principal objetivo fue perfeccionar el proceso de desarrollo de software mejorado por medio de la metodología con base a un desarrollo híbrido, así como la ejecución de pruebas integrales para reducir el tiempo de término de los sistemas de TI. La metodología utilizada fue la acción participativa, incluyéndose participantes como agentes activos con la intención de identificar aspectos relevantes como modelos existentes para la ejecución mediante procedimientos específicos de gestión de requisitos de software, gestión del proceso para desarrollar software y gestión de test para probar funcionamiento del software. En los resultados se brindaron propuestas y soluciones sobre los requerimientos del usuario mediante el proceso de desarrollar software alineando metodologías para desarrollar software de manera tradicional. Los autores concluyen que mediante la propuesta se logró integrar de manera oportuna los sistemas informáticos, lo cual mejoró la gestión frente a posibles cambios y circunstancias específicas del entorno.

En la investigación de Carhuaricra (2018) que fue publicado por el Repositorio de la Universidad César Vallejo. El principal objetivo fue identificar cómo influye un sistema web en el proceso de control de proyectos y aumentar el nivel de satisfacción del personal respecto a la gestión de proyectos. En los resultados se pudo conocer que al implementar una herramienta tecnológica permite al equipo de desarrollo incrementar los índices de rendimiento del cumplimiento del plan del trabajo y variación de los costos. Se concluye que hubo un aumento en el rendimiento del plan de trabajo, ya que el sistema permite planificar desde el inicio del procedimiento de control de los proyectos.

En la investigación de Céspedes (2017) que fue publicado por el Repositorio de la Universidad Peruana de Ciencias Aplicadas. El principal objetivo fue realizar

una mejora en el proceso de gestión de proyectos mediante la metodología Lean Project Management. En los resultados se pudo observar que la empresa pierde entre un 20 y un 35% del beneficio obtenido por sus ventas (proyectos) por el pago de una sanción contractual o sobrecostos porque la empresa tiene que ejecutar un plazo más amplio de entrega o también se disponen a contratar un número mayor de personal para corregir defectos. Como tal, es una inversión que se amortiza en el corto plazo (aproximado 6 meses) con el fin de contar con herramientas de control y aseguramiento de la calidad. Se concluye que al realizar un modelo de mapeo de procesos hubo mejoras que incluyen definir y analizar necesidades previas a la producción de módulos de información y garantizar la calidad utilizando KPIs según las bases de la metodología Lean Project Management

En la investigación de Espinoza y Espinoza (2017) que fue publicado por el Repositorio de la Universidad Peruana de Ciencias Aplicadas. El principal objetivo fue implementar un producto software mediante el uso de un marco de trabajo de design thinking y metodologías para mejorar el software. La metodología fue cuasi experimental, en la que se utilizó fases e ideas en el proceso de recolección de necesidades y en el proceso de diseñar el software. En los resultados se logró agilizar los softwares, lo cual permite identificar el nivel del usuario final y su aporte en el proyecto. Los autores concluyen que mediante la aplicación del design thinking, se logró aumentar de manera significativa el involucrar al usuario en el proyecto.

Moura et al. (2018), señala que un proyecto es un procedimiento exclusivo, que cuenta de una secuencia de pasos debidamente coordinados y controlados, que cuentan con una fecha de inicio y finalización realizado para lograr el objetivo de validar requisitos específicos, incluidas las limitaciones de tiempo y recursos. La naturaleza fundamental de un proyecto es que se tiene que ver con la realización de un esfuerzo que cuenta con un tiempo determinado para la elaboración de un producto, servicio o resultado particular, por ello, los proyectos se distinguen de las operaciones y de los programas.

Por su parte, Ariza (2017) señala que los proyectos son los que generan productos y servicios de software que brindan funciones de información

específica y de gestión, siendo caracterizados por su complejidad al aplicarse en todas las áreas del conocimiento, por el desarrollo continuo de tecnología y por muchos cambios existentes presentados en su ciclo de vida. Teniendo en cuenta lo anterior, para gestionar la amplitud o alcance de un proyecto, se requiere de una gestión durante toda la duración del proyecto. No puede ser simplemente iniciado o planeado y dejado solo; debe planificarse continuamente, supervisado y controlado. Basándose en ello, una gestión de proyectos es indicativa de planificación, realización de un seguimiento y realizar controles de proyectos incompletos o ausentes.

La gestión de proyectos es definida como un proceso de aplicar conocimiento, habilidad, herramienta y técnica a la actividad realizada en el proyecto que cumplen con los requerimientos del proyecto; es decir, la gestión de proyectos es un grupo de procesos interrelacionados que permite que el proyecto sea exitoso (Moura et al.,2018). Cabe mencionar que, el éxito de la gestión de proyectos se mide utilizando criterios técnicos tangibles tales como medidas variables, en las que se incluye cuatro dimensiones: tiempo, costo, alcance y calidad. Estos criterios técnicos forman la base de la investigación relacionada con el éxito del proyecto.

Asimismo, Bayona et al. (2016) señala que, como objetivo la gestión de proyectos canaliza de forma eficiente y productiva el trabajo de los desarrolladores, de tal forma que se conduce al éxito del proyecto, pues, dicho éxito incluye la gestión activa de los procesos con la intención de cumplir de manera exitosa con requisitos interesados.

Por su parte, Ramos et al. (2017) menciona que, gestionar un proceso de desarrollo de software mediante modelos, procesos, actividades o herramientas específicas, tiene sentido en el contexto de una empresa u organización, por lo que, es fundamental alinear las necesidades de la entidad basándose en el producto de software con la finalidad de gestionar actividades de desarrollo, control de tiempo, costos y calidad para la eficiencia del proyecto. En efecto, la gestión de proyectos es cada vez más importante para las organizaciones porque los proyectos son el método por el cual las organizaciones responden a su entorno. Siendo importante dentro de la gestión de proyectos son los

estándares y métodos que se utilizan para controlar y llevar a cabo proyectos, conocidos colectivamente como métodos de gestión de proyectos.

Un proceso de desarrollo de software es un conjunto de actividades y resultados relacionados que forman un software, centrándose en aspectos técnicos como la especificación, desarrollo, validación y evolución del software y, a la vez, proporciona transparencia y flexibilidad para facilitar la gestión del proyecto (Ramos et al., 2017). De modo similar, Carranza (2016) menciona que un proceso de software facilita un entorno de trabajo, estableciéndose un procedimiento detallado para el desarrollo del software o aplicación de un servicio. En ese sentido, las organizaciones deben elegir el modelo o estándar que mejor se adapte a su sistema de trabajo, ya que, según Ramos et al. (2017), los procesos de desarrollo pueden incluir actividades de gestión, tales como el proceso de gestión, pues es posible acoger una combinación de procesos complementarios de acuerdo con lo que requiere el proyecto.

En efecto, el desarrollo de software es una tarea muy compleja, requiere comunicación con los clientes, definir tareas y relaciones, entre otros; es decir, se debe llevar a cabo un plan para desarrollar el software. Esto es afirmado por Bayona et al. (2016) quien manifiesta que el desarrollo de software requiere un aumento de productividad y aseguramiento de la calidad del producto y cualquier proceso involucrado, así como su control a través de la planificación, estimación, controles y el seguimiento a las actividades en ejecución.

El ciclo de vida del desarrollo de software es uno de los problemas clave en el ingeniero de software, esto porque la estructura de actividades necesarias para desarrollar un sistema de software pesa sobre la calidad, utilidad y usabilidad del software desarrollado. Saeedi & Visvizi (2021), señalan que el ciclo de vida del desarrollo de software consta de cinco fases que incluyen: análisis de requisitos, diseño de software, implementación de software y pruebas de software. La fase de análisis de requisitos es el proceso de capturar los requisitos del usuario y la producción; durante la fase de diseño del software, los modelos de software abstractos se desarrollan a través de la ilustración de diferentes perspectivas utilizando lenguajes de modelado como el lenguaje de modelado unificado. Asimismo, la fase de implementación del software es el proceso de construcción

del software ejecutable basado en modelos construidos en la fase de diseño anterior. Finalmente, la fase de prueba de software está destinada a garantizar que el software desarrollado cumpla con las especificaciones anteriores y sin errores.

Halvorsen (2020), indica que, actualmente existen muchos procesos o métodos de desarrollo de software diferentes en uso, dividiéndose en 2 categorías principales: modelos basados en planes y métodos ágiles. Los métodos tradicionalmente basados en planes se usaban en el desarrollo de software; sin embargo, los métodos ágiles como Scrum, se han vuelto muy populares, especialmente en equipos de desarrollo más pequeños.

Los métodos convencionales incluyen un proyecto con un gran todo y una estructura definida, un proceso secuencial con un solo sentido y sin retroceso, un proceso rígido y sin cambios, porque las necesidades son inmediatas y para todo el proyecto, requiere una larga planificación previa y poca comunicación con los clientes una vez finalizado el proceso (Molina et al., 2018). Los métodos tradicionales de desarrollo de software son los modelos más antiguos y aún de uso común en muchas organizaciones, lo que conlleva a implementar un patrón similar que consiste en establecer etapas bien definidas. En este tipo de métodos, no se enfatiza la comunicación frecuente con los clientes y buena respuesta al cambio de requisitos.

Uno de los métodos tradicionales es el método en cascada, definida como una metodología de desarrollo de software impulsada por un plan basado en planificación avanzada de todas las fases del proyecto, que posteriormente se implementan. Este enfoque para el desarrollo de software es útil en la coordinación de trabajo de un gran proyecto entre diferentes equipos. No obstante, el inconveniente de la cascada es que es difícil acomodar los cambios necesarios, las correcciones y/o ajustes posibles y necesarios a medida que se desarrolla el proyecto y sus fases (Dima y Maassen, 2018).

En cierto modo, un cierto grado de dependencia de la trayectoria es inherente al método de la cascada. De hecho, los comentarios del cliente se reciben al final de todo el proceso, lo que se suma al desafío de personalizar el software desarrollado. En este punto de vista, el método de cascada es un método

adecuado en todos aquellos casos en los que todos los requisitos funcionales y no funcionales son claros, bien comprendidos y previsibles (Jinjin & Berlin, 2020). Por otra parte, Abdalhamid y Mishra (2017), señalan que el método ágil o incremental, se basa en una entrega incremental de funciones de software, es decir, el software se construye paso a paso, por lo que, en cada una de las etapas posteriores de la implementación del proyecto, se evalúa el resultado del proceso en contra de los comentarios recibidos del cliente. En otras palabras, a diferencia de la cascada, la retroalimentación se introduce en el proceso de forma instantánea e incremental, en lugar de solo al final del proceso.

Islam y Ferworn (2020), indican que el método ágil requiere trabajo en equipo avanzado y habilidades relacionadas, puesto que, el flujo de trabajo se divide en unidades de trabajo encuadradas en el tiempo durante las cuales se va a realizar un subcomponente específico del proyecto entregado. Los entregables se priorizan de acuerdo con su valor comercial, es decir, esencialmente, según las especificaciones y necesidades del cliente. Si todo el trabajo planeado para una determinada unidad de trabajo no se puede completar, el trabajo se vuelve a priorizar y la información se utiliza para el futuro planificación de trabajo.

Se debe agregar que, el método ágil se propuso por primera vez como una alternativa al desarrollo de software existente. La palabra ágil se asocia a un conjunto de valores y principios sobre los cuales se construye el método. Un proyecto comienza con abordar los requisitos de mayor prioridad del cliente en un trabajo software, sigue la evaluación del cliente de los requisitos desarrollados y se adapta a la fecha correspondiente de retroalimentación del cliente. En este enfoque, el cliente se beneficia desde el software ya en la etapa inicial del proyecto, que es esencial en vista de adelantarse a posibles errores/fallos y desajustes en requisitos de usuarios. En ese sentido, el enfoque incremental y la participación activa del equipo y del cliente fomentan la confianza entre ellos, siendo esto importante para el éxito del proyecto (Sawalha y Nabi, 2020). De lo anterior, se menciona que las metodologías ágiles describen una serie de principios que en resumen ponen al factor humano (clientes y desarrolladores) primero sobre los procesos y planes, pues, su principio de primacía es satisfacer al cliente mediante la entrega de entrega continua de software. Además, se

apoya en el enfoque descentralizado con respecto a la gestión y a elegir la mejor decisión, hacer y motivar a todas las personas a dar lo mejor de sí mismas para lograr el objetivo central a tiempo, teniendo en cuenta el escenario dinámico del mercado actual. En cada iteración, se intenta proporcionar la máxima cohesión y el mínimo acoplamiento entre los diferentes submódulos que se van integrando en el futuro.

Por consiguiente, el desarrollo de software ágil o las metodologías ágiles son un grupo de procesos de desarrollo de software que se trata de realizar un desarrollo incremental o el desarrollo iterativo. Los requisitos y las soluciones del proceso de desarrollo ágil se cambian y evolucionan a través de colaboración y comunicación entre organización y equipos (Mohammad et al., 2013). La concentración principal se da a las personas (cliente / usuarios finales o miembro del equipo). Software ágiles procesos son capaces de lidiar con cambios rápidos desde el lado del usuario o debido al entorno cambiante efectos (Pócsová et al., 2020).

Los métodos de desarrollo ágil más notables incluyen: Scrum (muy popular entre los startups), Extreme Programming (XP) Crystal Clear, Mobile-D (ágil y extremo para dispositivos móviles), Adaptive Software Development (ASD), Lean Development. Se señala que Scrum y XP se encuentran actualmente entre los más populares.

En definitiva, los métodos de desarrollo tradicionales asignan una disciplina de trabajo basada en documentar el proceso de desarrollo del software y enfatizando en la planificación integral e integral de todo el trabajo a realizar, pues después del detalle el ciclo comienza con el desarrollo del software. Sin embargo, en las metodologías de desarrollo ágiles, la documentación generalmente se ignora y se enfoca en el trabajo para encontrar un equilibrio entre el proceso y el esfuerzo.

Cabe precisar que, la práctica de los proyectos de desarrollo de software sugiere que los elementos, tanto de métodos incrementales como los basados en planes, se aplican todo el desarrollo del software. Sin embargo, se afirma que más del 70% de las organizaciones utilizan el desarrollo de software de método ágil en el

que la entrega de sub componentes funcionales de todo el proyecto es colocada en el desarrollo (Saeedi y Visvizi, 2021).

Emran et al. (2020), señala que tanto los procesos de ingeniería de software como las técnicas de gestión de calidad tienen fallas si no se basan en el conocimiento necesario o no son producidos dentro de la organización. Es por ello que, al desarrollar proyectos de software se genera experiencia en los desarrolladores que se acumulan en las mentes de los desarrolladores como conocimiento individual y colectivo en la organización, el cual se debería compartir para evitar nuevas gestiones en cada nuevo proyecto y organización.

Respecto a ello, Mighetti y Hadad (2016) manifiestan que el objetivo de la gestión del conocimiento es recopilar todo el conocimiento de la organización para su posterior uso, incluida la información sobre los proyectos pasados y actuales. Este elemento es especialmente importante en un paradigma distribuido, en el que los actores están dispersos geográficamente y requieren un acceso formal a la información que se va a comunicar; esto impide el conocimiento tácito y la transmisión informal de información.

Además, los gerentes de proyecto deben enfocarse en comprender y atender las necesidades de comunicación de cada persona involucrada en el proyecto, lo cual implica determinar la comunicación e impacto en el transcurso del proyecto. Esto incluye actualizaciones de estado del proyecto, minutos de reuniones e informes sobre entregables, junto con otra documentación del proyecto (Ferrer, 2018). Lo mencionado indica que es fundamental la parte comunicativa, durante el desarrollo de cada actividad dentro de todo el proceso para desarrollar software.

III. MÉTODO

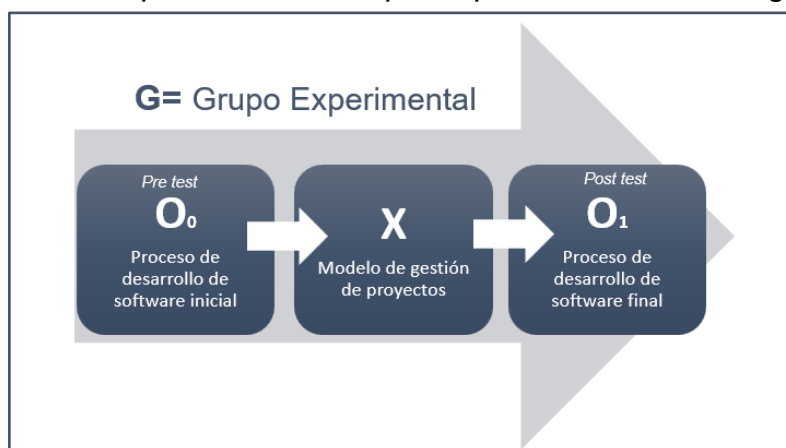
3.1. Tipo y diseño de investigación

Según Nieto (2018) un estudio de tipo aplicada tiene como objetivo mejorar, perfeccionar y optimizar las funciones de los sistemas, procedimientos, estándares y reglas técnicas vigentes en base a los avances de la tecnología y la ciencia; por lo tanto, este tipo de investigación no aplica a restricciones verdaderas, falsas o posibles, sino a límites eficientes, defectuosos e ineficientes, válido o inválido. Por lo expuesto, esta investigación tiene un tipo aplicado.

En esta investigación se adoptó por un enfoque cuantitativo, debido a que los datos recogidos eran medibles, ya que se introdujeron en una base de datos para su posterior interpretación, y era cuantificable porque los datos obtenidos permitían al investigador descubrir una conexión entre las dos variables investigadas (Hernández, et al, 2014).

Para Freire (2018) se considera una investigación de diseño experimental a momento cuando se aplica una alteración en el ambiente natural de la variable que es dependiente debido a que se manipula la variable independiente, en el presente estudio se plantea es un modelo de gestión que permite mejorar el proceso de desarrollo de software, por ello la investigación es experimental en el grado pre experimental, con la aplicación de una prueba de entrada y salida. En relación a lo mencionado con anterioridad, el estudio se basa en el esquema mostrado a continuación:

Figura 1. Esquema de diseño pre experimental de investigación



Fuente: Elaboración propia

Donde:

G: Grupo Experimental

O1: Proceso de desarrollo de software inicial

X: Modelo de gestión de proyectos

O2: Proceso de desarrollo de software final

3.2. Variables y operacionalización

Variable independiente: Modelo de gestión de proyectos. - Un modelo de gestión de proyectos es un conjunto de reglas y recomendaciones para llevar a cabo cada tarea durante el desarrollo del proyecto. Para que sea más fácil intercambiar información para los integrantes del equipo, el modelo de proyecto suele representarse gráficamente (Pérez, 2021).

Dimensiones: Planificación, Desarrollo y Cierre

Variable dependiente: Proceso de desarrollo de software. – se denomina al proceso para desarrollar software como un grupo de recurso humano, modelos organizativos, normas, políticas, ejercicios y procedimientos, elementos de software, procesos y herramientas que se utilizan o crean para definir, desarrollar, entregar, innovar y ampliar un producto de software (Ruvalcaba, 2020).

Dimensiones: Curva de Aprendizaje, Tiempo de desarrollo, Comunicación con el usuario y Aceptación del usuario final.

3.3. Población, muestra y muestreo

Población: Una población determinada para un estudio está formada por varios elementos que se usaron en la investigación, todos ellos con características similares. La investigación estadística se realiza en esta población con el objetivo de completar el estudio (Arias et al., 2016). La población es equivalente al total de trabajadores encargados de los Proyectos de Software de la Corte Superior de Justicia 25 trabajadores.

Muestra: La muestra es considerada un grupo que forma parte de la población total, que está formado por componentes con cualidades similares, determinada a partir de un muestreo, los cuales son partícipes

directos de la aplicación del estudio para la recolección de datos (Arispe et al., 2020). La población por ser una cantidad menor, la muestra fue igual a la población, es decir se tomó como estudio a los 25 colaboradores de la Corte Superior de Justicia de Lambayeque.

Muestreo: Dado que el 100% de la población participara en el estudio, el muestreo ya no es aplicable en este caso.

3.4. Técnicas e instrumentos de recolección de datos

Técnica de recolección de datos: Una técnica es el grupo de actividades empleadas con el objetivo de recolectar datos para obtener información que se desea en un ambiente o zona definida (Hernández, et al, 2017).

En este estudio, la técnica empleada fue la encuesta, que es aplicada a la muestra para poder encontrar una relación entre las variables investigadas.

Después se aplicó la técnica de recolección de datos mediante cuestionarios o encuestas que permitió recolectar información con respecto al nivel de cumplimiento de comunicación entre las áreas involucradas.

Así como la técnica de la observación utilizada en las fichas de registro, lo que permite recolectar datos significativos para la investigación, que fueron recolectadas con anterioridad a la implementación y posterior a la implementación del modelo de gestión de proyectos.

Instrumento de recolección de datos: Sirven para el proceso de recolección de información donde se pueden utilizar formularios, registros, dispositivos validados o de elaboración propia, ya que permiten recibir toda la información obtenida tras su aplicación (Páramo y Gómez, 2008).

En este estudio se hizo uso como instrumento al cuestionario, ya que se aplica una encuesta al gerente para conocer la situación de la empresa con un total de 8 preguntas y a los usuarios, el primer cuestionario cuenta con un total de 16 preguntas, con alternativas dicotómicas (SI y NO) destinada a medir el nivel de conocimiento sobre metodologías ágiles, el segundo cuestionario cuenta con un total de 5 categorías, con alternativas tipo Likert desde 1 (pésimo) a 4 (excelente) para cada categoría, que ayudará a medir el nivel de satisfacción de los usuarios al finalizar el desarrollo del software.

Tabla 1. Técnicas e instrumentos de investigación.

Técnica	Instrumento	Fuente
Encuesta	Cuestionario	Coordinador de informática
Recolección de datos	Ficha de registros de base de datos	Áreas usuarias

Fuente: Elaboración propia

Validez: Para la realización de la validez es primordial contar con una evaluación a nivel de cohesión interna y nula contradicción en relación al resultado que se obtengan (Páramo y Gómez, 2008).

Por lo mencionado anteriormente, a través de la participación de 3 expertos profesionales en el ámbito de estudio, quienes realizaron una evaluación respectiva, realizaron la validación por medio de la matriz presentada que relaciona ambas variables investigadas, por medio de indicadores y respetando las dimensiones de cada variable.

Tabla 2. Tabla Índice de Validez de Contenido (Lawshe).

Número del ítem	JUECES			NO ESENCIAL		CVR	DECISIÓN	
	1	2	3	ESENCIAL	ESENCIAL			
1	1	1	1	3	3	0	1	Incluir
2	1	1	1	3	3	0	1	Incluir
3	1	1	1	3	3	0	1	Incluir
4	1	1	1	3	3	0	1	Incluir
5	1	1	1	3	3	0	1	Incluir
6	1	1	1	3	3	0	1	Incluir
7	1	1	1	3	3	0	1	Incluir
8	1	1	1	3	3	0	1	Incluir
9	1	1	1	3	3	0	1	Incluir

10	1	1	1	3	3	0	1	Incluir
11	1	1	1	3	3	0	1	Incluir

Fuente: base de datos

En la tabla 1 se aprecia los valores para los criterios del índice de validez de contenido, se observa que los ítems analizados, son de importancia, por lo cual no se llegaría a excluir ninguno, debido a que todos tienen una razón de validez de 1.

Confiabilidad: De acuerdo con lo mencionado por Hernández, et al (2014), al hablar de la confiabilidad de un determinado instrumento, se hace referencia al nivel de aplicabilidad que este tiene en una muestra similar, generando así resultados coherentes y consistentes.

En la presente investigación, se hizo uso del cuestionario, se realizaron 2 formularios las cuales fueron sometidas a una prueba piloto, en el primer instrumento realizado para el objetivo 1: Determinar el nivel de conocimiento sobre metodologías ágiles, se utilizó el coeficiente de Kuder-Richardson para determinar su confiabilidad.

Tabla 3. *Valores de consistencia interna mediante el coeficiente Kuder-Richardson*

Kr - 20	N de elementos
0,775	20

Fuente: base de datos

En relación a los resultados de la tabla anterior, se evidencia que la confiabilidad del instrumento resultó tener un coeficiente muy bueno, proporcionando un valor de 0.775. Con estos resultados se puede concluir que la confiabilidad del instrumento es del 77.5%.

En el segundo instrumento realizado para el objetivo 4: Determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la Corte Superior de Justicia, se empleó el coeficiente de McDonald's Omega para poder determinar su confiabilidad.

Tabla 4. *Valores de consistencia interna mediante el coeficiente McDonald's*

McDonald's Omega	N de elementos
0,844	5

Fuente: base de datos

En relación con los resultados mostrados en la tabla anterior, se puede evidenciar que la confiabilidad del instrumento resultó tener un coeficiente muy bueno, la cual proporcionó un valor de 0.844. Con estos resultados se puede concluir que la confiabilidad del instrumento es del 84.4%.

3.5. Procedimientos

Esta investigación empieza con la realización de una entrevista hacia el coordinador de informática, el cual contestó una serie de interrogantes que permitió conocer la situación actual que se viene desarrollando en el ambiente organizacional de la Corte Superior de Justicia de Lambayeque. Por otro lado, es necesario también conocer las diferentes funciones que realizan los colaboradores del área de informática lo cual permitió conocer con más profundidad la gestión de proyectos desde diferentes roles del área.

Posteriormente, se procede con la formulación y elaboración del planeamiento de la problemática actual de la Corte Superior de Justicia, con la definición del objeto a estudiar y el alcance que tendrá el estudio.

Al conocer la problemática, se procede a la definición del título del estudio, hipótesis y objetivos tanto generales como específicos que permitirá evaluar en qué punto estamos del proyecto y tener un control de lo que queremos llegar.

De la misma manera, se identifica la población de estudio donde están involucrados tanto los colaboradores de informática y el área usuaria de los proyectos informáticos.

Con la finalidad de conocer la metodología, teorías y herramientas que se utilizarán en la investigación se realiza el marco teórico de la investigación

con ayuda de otras investigaciones que permite verificar los antecedentes del proyecto.

Para poder conocer más al detalle se realiza la descripción de instrumentos para la recolección de datos como encuestas y entrevistas.

Luego se procedió a realizar el pre test, donde se procedió aplicar la recolección de datos con el sustento de nuestros indicadores ya definidos donde se identificó el tiempo de demora que se da durante la elaboración de un proceso de software (Anexo 3.3), después se procederá a determinar el nivel de comunicación entre las áreas involucradas para el desarrollo de software, el cual determinará cuántas veces se reunieron las áreas en el tiempo planificado. (Anexo 3.4)

Para poder aplicar una metodología ágil, en los procesos de desarrollo de software primero se realizó un cuestionario para poder evaluar el nivel de conocimiento de los colaboradores de la Corte Superior de Justicia del área de Informática (Anexo 3.2)

Con respecto a poner en marcha del sistema de gestión de proyectos, podrá apoyar en la planificación, monitoreo y captura de requerimientos de los distintos proyectos informáticos donde finalmente podremos medir el nivel de satisfacción del área usuaria.

Cabe señalar que la Corte Superior de Justicia autorizó poder aplicar estos instrumentos, ya que, proporciona al investigador la documentación de aceptación para la realización y aplicación del estudio, como se puede visualizar en (Anexo 5).

Con la información de los datos recolectados se procederá a la realización de un análisis estadístico, por lo que será necesario utilizar los softwares como Microsoft Excel, IBM SPSS Software. Con el uso de estas herramientas se podrá visualizar información que muestre una mejor visión de la situación actual del problema respecto a los indicadores mencionados.

Posterior a ello, se aplicará la gestión de proyectos para la mejora del proceso de desarrollo de software dentro de la corte superior de justicia de Lambayeque, una vez aplicada la mejora correspondiente se aplicará la

evaluación de los indicadores iniciales, en post de determinar la mejora obtenida.

3.6. Método de análisis de datos

Los datos serán recolectados por medio de un cuestionario luego, se procesarán en hojas de cálculo en Microsoft Excel, para la realización del procesamiento estadístico haciendo uso del software SPSS v.23, donde se efectuará la prueba de normalidad para determinar el estadístico de prueba Wilcoxon y así, presentar la data a través de gráficos y tablas, las cuales se interpretarán y analizarán, verificando si es que se cumplió con las hipótesis establecidas. Asimismo, se realizará una comparativa, con la finalidad de verificar si es que el modelo para la gestión de proyectos en realidad mejora el proceso de desarrollo de Software de la Corte Superior de Justicia de Lambayeque.

Tabla 5. *Hipótesis, el nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.*

O1	Determinar el nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.
Indicador 1	Nivel de conocimiento de los equipos de desarrollo
Donde:	
NCMGa	Nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos antes de la implementación de un modelo de gestión de proyectos.
NCMGd	Nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos después de la implementación de un modelo de gestión de proyectos.
Hipótesis Nula H0	El modelo de gestión de proyectos disminuye el nivel de conocimiento de los equipos de desarrollo.

	H0: NCMGd – NCMGa \leq 0
Hipótesis Alternativa H1	El modelo de gestión de proyectos aumenta el nivel de conocimiento de los equipos de desarrollo. H1: NCMGd – NCMGa $>$ 0

Fuente: Elaboración propia

Tabla 6. Hipótesis, el tiempo promedio de entrega de desarrollo de software

O2	Determinar el tiempo promedio de entrega de desarrollo de software.
Indicador 2	Tiempo promedio de entrega de desarrollo de software.
Donde:	
TPESDa	Tiempo promedio de entrega de desarrollo de software antes de la implementación de un modelo de gestión de proyectos.
TPESDd	Tiempo promedio de entrega de desarrollo de software después de la implementación de un modelo de gestión de proyectos.
Hipótesis Nula H0	El modelo de gestión de proyectos no disminuye el tiempo promedio de entrega de desarrollo de software. H0: TPESDa – TPESDd \geq 0
Hipótesis Alternativa H1	El modelo de gestión de proyectos disminuye el tiempo promedio de entrega de desarrollo de software. H1: TPESDa – TPESDd $<$ 0

Fuente: Elaboración propia

Tabla 7. Hipótesis, el nivel de comunicación con el área usuaria durante el desarrollo del software.

O3	Determinar el nivel de comunicación con el área usuaria durante el desarrollo del software.
Indicador 3	El nivel de comunicación con el área usuaria durante el desarrollo del software.
Donde:	
NCAUa	Nivel de comunicación con el área usuaria antes de la implementación de un modelo de gestión de proyectos.
NCAUd	Nivel de comunicación con el área usuaria después de la implementación de un modelo de gestión de proyectos.
Hipótesis Nula H0	El modelo de gestión de proyectos no aumenta el nivel de comunicación con el área usuaria durante el desarrollo del software. H0: $NCAUa - NCAUd \geq 0$
Hipótesis Alternativa H1	El modelo de gestión de proyectos aumenta el nivel de comunicación con el área usuaria durante el desarrollo del software. H1: $NCAUa - NCAUd < 0$

Fuente: Elaboración propia

Tabla 8. *Hipótesis, el nivel de satisfacción del usuario al finalizar el desarrollo del software*

O4	Determinar el nivel de satisfacción del usuario al finalizar el desarrollo del software.
Indicador 4	El nivel de satisfacción del usuario al finalizar el desarrollo del software.
Donde:	
NSUFDSa	Nivel de satisfacción del usuario al finalizar el desarrollo del software antes de la implementación de un modelo de gestión de proyectos.

NSUFDSd	Nivel de satisfacción del usuario al finalizar el desarrollo del software después de la implementación de un modelo de gestión de proyectos.
Hipótesis Nula H0	El modelo de gestión de proyectos disminuye el nivel de satisfacción del usuario al finalizar el desarrollo del software. H0: NSUFDSa – NSUFDSd ≥ 0
Hipótesis Alternativa H1	El modelo de gestión de proyectos aumenta nivel de satisfacción del usuario al finalizar el desarrollo del software H1: NSUFDSa – NSUFDSd < 0

Fuente: Elaboración propia

3.7. Aspectos éticos

La investigación tiene en consideración a las siguientes condiciones éticas, planteadas en las normas y artículos de la Resolución del consejo Universitario N° 00126-2017-UCV. Que se encuentran en razón al Art.14 con respecto a publicar los estudios de investigación, se hará una autorización que asegura la veracidad de esta investigación aceptando comprometerse ética y moralmente. En el Art.15 sobre las Políticas anti plagio, esta investigación se evaluará a través del software turnitin. Art.16 que se centra en el respeto de los Derechos de autor, realizándose una declaración de validez única y sin cometer plagio y con respeto al Art.15 de la Resolución del Consejo Universitario N° 00126-2017-UCV. El Art. 17 del autor principal y personal que investiga, debido a que como investigadores nos responsabiliza a preservar la verdad y validez del resultado, así como de la confiabilidad aplicada a los instrumentos otorgados por la organización

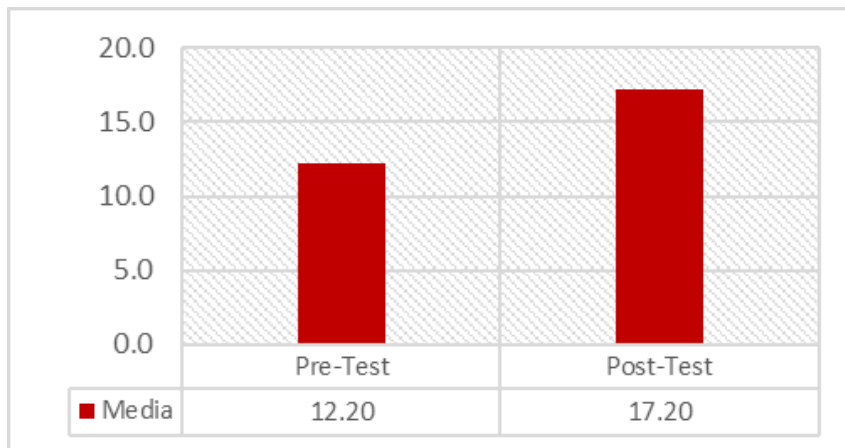
IV. RESULTADOS

Tabla 9. Estadísticos descriptivos del indicador nivel de conocimiento sobre metodologías ágiles

	N	Media	Desv. Desviación	Mínimo	Máximo
Pre-Test	25	12.20	2.000	8	15
Post-Test	25	17.20	1.258	15	20

Fuente: base de datos

Figura 2. Promedio del nivel de conocimiento sobre metodologías ágiles



Fuente: base de datos

En relación a los resultados analizados en el indicador para determinar el nivel de conocimiento sobre metodologías ágiles, se evidenció que el pre test en promedio es del 12.20%, mientras que en el post test se evidencia un aumento en promedio del 17.20%.

H₀: Los datos de las variables presentan distribución normal.

H₁: Los datos de las variables no presentan distribución normal.

Tabla 10. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test del cuestionario de nivel de conocimiento sobre metodologías ágiles.

	Shapiro-Wilk		
	Estadístico	gl	Sig.
Pre-Test	0,920	25	0,051
Post-Test	0,905	25	0,023

Fuente: base de datos

En base a la tabla anterior se evidencia que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles Pre Test, analizada con la prueba de Shapiro-Wilk resultó tener un nivel de significancia mayor a 0.05, teniendo un valor de 0.051, lo cual indica que el Pre-Test proviene de una distribución normal, Así mismo para Post-Test, analizada con la prueba Shapiro Wilk, arrojó un nivel de significancia menor a 0.05, teniendo un valor de 0.023, la cual nos indicaría que la variable no proviene de una distribución normal. Se llega a la conclusión que para medir la relación entre el Pre-Test y el Post-Test se desarrolló una prueba no paramétrica.

H₀: El modelo de gestión de proyectos no mejora el nivel de conocimiento sobre las metodologías ágiles.

H₁: El modelo de gestión de proyectos mejora el nivel de conocimiento sobre las metodologías ágiles.

Tabla 11. Rangos obtenidos en el nivel de conocimiento sobre metodologías ágiles antes y después.

		N	Rango promedio	Suma de rangos
Post-Test - Pre-Test	Rangos negativos	0 ^a	0,00	0,00
	Rangos positivos	25 ^b	13,00	325,00
	Empates	0 ^c		
	Total	25		

a. Post-Test < Pre-Test

b. Post-Test > Pre-Test

c. Post-Test = Pre-Test

Tabla 12. *Estadísticos de prueba*

	Post-Test - Pre-Test
Z	-4,382 ^b
Sig. asintótica(bilateral)	0,000

a. Prueba de rangos con signo de Wilcoxon

b. Se basa en rangos negativos.

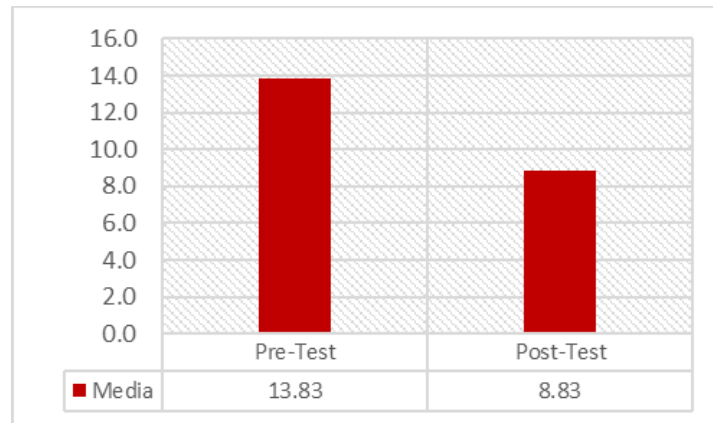
En la tabla de Análisis del cuestionario Pre y Post Test para determinar el nivel de conocimiento sobre metodologías ágiles, vemos que se analizaron 25 pares (los 25 empleadores). Hubo 0 rangos negativos, 25 positivos y 0 empates. Así mismo en la siguiente tabla sobre los “Estadísticos de prueba” se puede visualizar la fila Sig. asintót. (bilateral) con un valor de 0,000. Entonces podemos afirmar que se rechaza la hipótesis nula, ya que el valor de p (sig. Asintót. (bilateral)) es inferior a 0,05. Llegando a la conclusión que existe evidencia competente para mencionar que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles es efectivo con un nivel de significación del 5%.

Tabla 13. *Estadísticos descriptivos del indicador tiempo promedio para la entrega del proyecto de software*

	N	Media	Desv. Desviación	Mínimo	Máximo
Pre-Test	6	13.83	2.041	11	17
Post-Test	6	8.83	2.041	7	11

Fuente: base de datos

Figura 3. Tiempo promedio para la entrega del proyecto de software.



Fuente: base de datos

En relación a los resultados analizados en el indicador para determinar el tiempo promedio para la entrega del proyecto de software, se evidenció que el pre test en promedio es del 13.83%, mientras que en el post test se evidencia una reducción en promedio del 8.83%.

H_0 : Los datos de las variables presentan distribución normal.

H_1 : Los datos de las variables no presentan distribución normal.

Tabla 14. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test de la ficha de registro de tiempo promedio para la entrega del proyecto de software.

	Shapiro-Wilk		
	Estadístico	gl	Sig.
Pre-Test	0,975	6	0,926
Post-Test	0,753	6	0,021

Fuente: base de datos

En base a la tabla anterior se evidencia que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles Pre Test, analizada con la prueba de Shapiro-Wilk resultó tener un nivel de significancia mayor a 0.05, teniendo un valor de 0.926, lo cual indica que el Pre-Test proviene de una distribución normal, Así mismo para Post-Test, analizada con la prueba Shapiro Wilk, arrojó un nivel de significancia menor a 0.05, teniendo un valor de 0.021, la cual nos indicaría que la variable no proviene de una distribución normal. Se llega a la conclusión que para medir la relación entre el Pre-Test y el Post-Test se desarrolló una prueba no paramétrica.

H₀: El modelo de gestión de proyectos no reduce el tiempo promedio de entrega de desarrollo de software.

H₁: El modelo de gestión de proyectos reduce el tiempo promedio de entrega de desarrollo de software.

Tabla 15. Rangos obtenidos en el tiempo promedio para la entrega del proyecto de software antes y después

		N	Rango promedio	Suma de rangos
Post-Test - Pre-Test	Rangos negativos	6 ^a	3,50	21,00
	Rangos positivos	0 ^b	0,00	0,00
	Empates	0 ^c		
	Total	6		

a. Post-Test < Pre-Test

b. Post-Test > Pre-Test

c. Post-Test = Pre-Test

Tabla 16. Estadísticos de prueba

	Post-Test - Pre-Test
Z	-2,214 ^b
Sig. asintótica(bilateral)	0,027

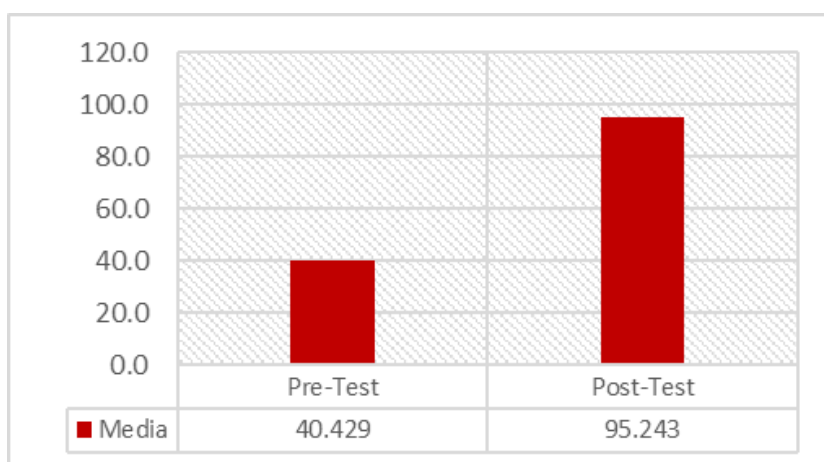
En la tabla de Análisis del cuestionario Pre y Post Test para determinar el nivel de conocimiento sobre metodologías ágiles, vemos que se analizaron 6 pares. Hubo 6 rangos negativos, 0 positivos y 0 empates. Así mismo en la siguiente tabla sobre los “Estadísticos de prueba” se puede visualizar la fila Sig. asintót. (bilateral) con un valor de 0,027. Entonces podemos afirmar que se rechaza la hipótesis nula, ya que el valor de p (sig. Asintót. (bilateral)) es inferior a 0,05. Llegando a la conclusión que existe evidencia competente para mencionar que la ficha de registro para determinar tiempo promedio para la entrega del proyecto de software es efectiva con un nivel de significación del 5%.

Tabla 17. Estadísticos descriptivos del indicador nivel de comunicación con el área usuaria

	N	Media	Desv. Desviación	Mínimo	Máximo
Pre-Test	7	40.429	44.9958	0.0	100.0
Post-Test	7	95.243	12.5862	66.7	100.0

Fuente: base de datos

Figura 4. Promedio del nivel de comunicación con el área usuaria



Fuente: base de datos

En relación a los resultados analizados en el indicador para determinar el nivel de comunicación con el área usuaria, se evidenció que el pre test en promedio es del 40.43%, mientras que en el post test se evidencia un aumento en promedio del 95.24%.

H_0 : Los datos de las variables presentan distribución normal.

H_1 : Los datos de las variables no presentan distribución normal.

Tabla 18. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test de la ficha de registro de nivel de comunicación con el área usuaria

	Shapiro-Wilk		
	Estadístico	gl	Sig.
Pre-Test	0,818	7	0,061
Post-Test	0,453	7	0,000

Fuente: base de datos

En base a la tabla anterior se evidencia que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles Pre Test, analizada con la prueba de

Shapiro-Wilk resultó tener un nivel de significancia mayor a 0.05, teniendo un valor de 0.061, lo cual indica que el Pre-Test proviene de una distribución normal, Así mismo para Post-Test, analizada con la prueba Shapiro Wilk, arrojó un nivel de significancia menor a 0.05, teniendo un valor de 0.000, la cual nos indicaría que la variable no proviene de una distribución normal. Se llega a la conclusión que para medir la relación entre el Pre-Test y el Post-Test se desarrolló una prueba no paramétrica.

H₀: El modelo de gestión de proyectos no aumenta el nivel de comunicación en el desarrollo de software.

H₁: El modelo de gestión de proyectos aumenta el nivel de comunicación en el desarrollo de software.

Tabla 19. Rangos obtenidos en el nivel de comunicación con el área usuaria antes y después

		N	Rango promedio	Suma de rangos
Post-Test - Pre-Test	Rangos negativos	0 ^a	0,00	0,00
	Rangos positivos	5 ^b	3,00	15,00
	Empates	2 ^c		
	Total	7		

a. Post-Test < Pre-Test

b. Post-Test > Pre-Test

c. Post-Test = Pre-Test

Tabla 20. Estadísticos de prueba

	Post-Test - Pre-Test
Z	-2,060 ^b
Sig. asintótica(bilateral)	0,039

a. Prueba de rangos con signo de Wilcoxon

b. Se basa en rangos negativos.

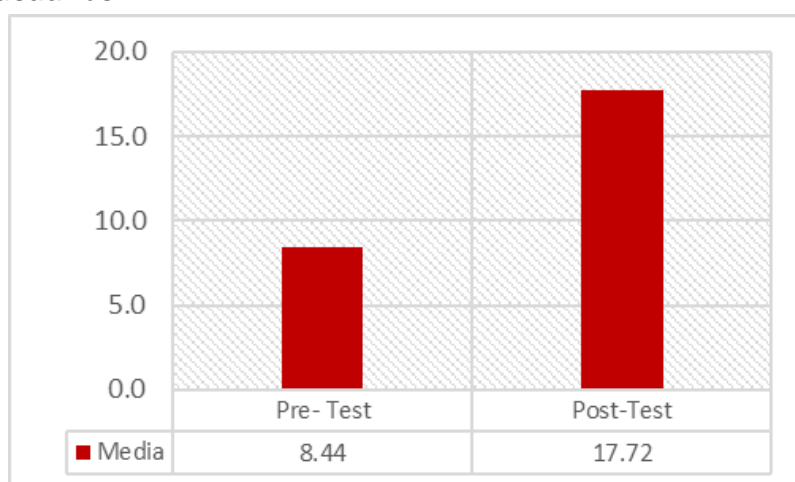
En la tabla de Análisis del cuestionario Pre y Post Test para determinar el nivel de conocimiento sobre metodologías ágiles, vemos que se analizaron 7 pares. Hubo 0 rangos negativos, 5 positivos y 2 empates. Así mismo en la siguiente tabla sobre los “Estadísticos de prueba” se puede visualizar la fila Sig. asintót. (bilateral) con un valor de 0,039. Entonces podemos afirmar que se rechaza la hipótesis nula, ya que el valor de p (sig. Asintót. (bilateral)) es inferior a 0,05. Llegando a la conclusión que existe evidencia competente para mencionar que la ficha de registro para determinar el nivel de comunicación con el área usuaria es efectiva con un nivel de significación del 5%.

Tabla 21. Estadísticos descriptivos del indicador nivel de satisfacción de los usuarios al finalizar el desarrollo del software

	N	Media	Desv. Desviación	Mínimo	Máximo
Pre- Test	25	8.44	2.274	5	12
Post-Test	25	17.72	1.514	15	20

Fuente: base de datos

Figura 5. Promedio del nivel de satisfacción de los usuarios



Fuente: base de datos

En relación a los resultados analizados en el indicador para determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior

de justicia, se evidenció que el pre test en promedio es del 8.44%, mientras que en el post test se evidencia un aumento en promedio del 17.72%.

H₀: Los datos de las variables presentan distribución normal.

H₁: Los datos de las variables no presentan distribución normal.

Tabla 22. Pruebas de normalidad Shapiro Wilk aplicado a la puntuación del Pre y Post Test del cuestionario de nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia.

	Shapiro-Wilk		
	Estadístico	gl	Sig.
Pre- Test	0,926	25	0,072
Post-Test	0,903	25	0,022

Fuente: base de datos

En base a la tabla anterior se evidencia que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles Pre Test, analizada con la prueba de Shapiro-Wilk resultó tener un nivel de significancia mayor a 0.05, teniendo un valor de 0.072, lo cual indica que el Pre-Test proviene de una distribución normal, Así mismo para Post-Test, analizada con la prueba Shapiro Wilk, arrojó un nivel de significancia menor a 0.05, teniendo un valor de 0.022, la cual nos indicaría que la variable no proviene de una distribución normal. Se llega a la conclusión que para medir la relación entre el Pre-Test y el Post-Test se desarrolló una prueba no paramétrica.

H₀: El modelo de gestión de proyectos no aumenta el nivel de satisfacción del usuario al finalizar el desarrollo del software

H₁: El modelo de gestión de proyectos aumenta nivel de satisfacción del usuario al finalizar el desarrollo del software

Tabla 23. Rangos obtenidos en el nivel de satisfacción de los usuarios al finalizar el desarrollo del software antes y después.

		N	Rango promedio	Suma de rangos
Post-Test - Pre- Test	Rangos negativos	0 ^a	0,00	0,00
	Rangos positivos	25 ^b	13,00	325,00
	Empates	0 ^c		
	Total	25		

a. Post-Test < Pre- Test

b. Post-Test > Pre- Test

c. Post-Test = Pre- Test

Tabla 24. Estadísticos de prueba

	Post-Test - Pre- Test
Z	-4,385 ^b
Sig. asintótica(bilateral)	0,000

a. Prueba de rangos con signo de Wilcoxon

b. Se basa en rangos negativos.

En la tabla de Análisis del cuestionario Pre y Post Test para el indicador de determinar el nivel de conocimiento sobre metodologías ágiles. Podemos observar que se analizaron 25 pares. Además de 0 rangos negativos, 25 positivos y 0 empates. Así mismo en la siguiente tabla sobre los “Estadísticos de prueba” se puede visualizar la fila Sig. asintót. (bilateral) con un valor de 0,000. Entonces podemos afirmar que se rechaza la hipótesis nula, ya que el valor de p (sig. Asintót. (bilateral)) es inferior a 0,05. Llegando a la conclusión que existe evidencia competente para mencionar que el cuestionario para determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia es efectivo con un nivel de significación del 5%.

V. DISCUSIÓN

Los resultados obtenidos y los cuales se mostrarán a continuación evidencian la información obtenida mediante las fichas de registro aplicadas a proyectos de Mesa de Partes Virtual Externa y Sistema de Eventos Académicos. Además, de la aplicación de cuestionarios aplicados a 25 trabajadores encargados de los Proyectos de Software de la Corte Superior de Justicia, que se analizaron durante 7 semanas mediante un pre test y post test en relación a la implementación de un Modelo de gestión de proyectos. Los resultados a nivel general evidencian la mejora del proceso de desarrollo de software con la implementación de un modelo de gestión de proyectos.

En contraste con la investigación de Rodríguez (2016) en el autor concluye que en la recolección de información las tecnologías desarrolladas en las organizaciones no siguen una metodología que gestione proyectos de desarrollo de software, obteniendo soluciones incompletas e inestables. Lo cual permite ver la importancia de la implementación de metodologías en el proceso de gestión de proyectos. Además, como en el caso de la investigación de Céspedes (2017) en sus resultados se pudo observar que la empresa pierde entre un 20 y un 35% del beneficio obtenido por sus ventas (proyectos) por el pago de una sanción contractual o sobrecostos porque la empresa tiene que ejecutar un plazo más amplio de entrega o también se disponen a contratar un número mayor de personal para corregir defectos. Como tal, es una inversión que se amortiza en el corto plazo (aproximadamente 6 meses) con el fin de contar con herramientas de control y aseguramiento de la calidad. Se concluye que al realizar un modelo de mapeo de procesos hubo mejoras que incluyen el proceso de definir y analizar las necesidades previas a la generación de módulos de información y garantizar de la calidad utilizando KPIs según las bases de la metodología Lean Project Management

Conforme al primer objetivo específico de determinar el nivel de conocimiento sobre las metodologías ágiles de gestión de proyectos en la Corte Superior de Justicia. Para comprobar la existencia de una diferencia significativa entre los resultados del pre test y post test en base a un cuestionario, se empleó la prueba no paramétrica

de Wilcoxon, en donde se concluyó que para medir la relación entre el Pre-Test y el Post-Test se debe emplear una prueba no paramétrica. En relación a los resultados analizados en el indicador para determinar el nivel de conocimiento sobre metodologías ágiles, se evidenció que el pre test en promedio es del 12.20%, mientras que en el post test se evidencia un aumento en promedio del 17.20%, lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, concluyendo que hay evidencias suficientes para plantear que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles es efectivo con un nivel de significación del 5%.

De acuerdo con Castro et al. (2020) en su investigación denominada “Desarrollo de un software web para la generación de planes de gestión de riesgos de software”, la creación de un software web para planes de gestión permite conocer mejor las metodologías de desarrollo evolutivo incremental similar a la metodología SCRUM donde incluye diferentes etapas desde la especificación de requerimientos hasta la realización de test de prueba del software lo que permite visualizar los requisitos pueden ser cambiantes a lo largo del ciclo de vida del software, siendo la conclusión de la investigación.

Por lo tanto, siendo el método ágil una entrega incremental de funciones de software, es decir, el software se construye paso a paso, por lo que, en cada una de las etapas posteriores de la implementación del proyecto, se evalúan los resultados del proceso con los comentarios recibidos del cliente, donde la retroalimentación se introduce en el proceso de forma instantánea e incremental, en lugar de solo al final del proceso. Se evidencia la importancia de conocer el funcionamiento de una metodología para su uso e implementación en diferentes proyectos de desarrollo (Abdalhamid & Mishra, 2017).

Así mismo guarda relación con lo mencionado por Mighetti y Hadad (2016) quienes manifiestan que el objetivo de la gestión del conocimiento es recopilar todo el conocimiento de la organización para su posterior uso, incluida la información sobre los proyectos pasados y actuales. Este elemento es especialmente importante en un paradigma distribuido, en el que los actores están dispersos geográficamente y requieren un acceso formal a la información que se va a comunicar; esto impide el

conocimiento tácito y la transmisión informal de información. Haciendo énfasis además en la importancia de conocer la forma de trabajo dentro de una organización para su correcto desarrollo. Guardando relación con Emran et al. (2020), señala que tanto los procesos de ingeniería de software como las técnicas de gestión de calidad tienen fallas si no se basan en el conocimiento necesario o no son producidos dentro de la organización. Es por ello que, al desarrollar proyectos de software se genera experiencia en los desarrolladores que se acumulan en las mentes de los desarrolladores como conocimiento individual y colectivo en la organización, el cual se debería compartir para evitar nuevas gestiones en cada nuevo proyecto y organización.

Conforme al segundo objetivo específico de determinar el tiempo promedio para la entrega del software con el apoyo de una plataforma informática que permita planificar los requerimientos de la Corte Superior de Justicia. Para comprobar la existencia de una diferencia significativa entre los resultados del pre test y post test en base a una ficha de registro, se empleó la prueba no paramétrica de Wilcoxon, en donde se concluyó que para medir la relación entre el Pre-Test y el Post-Test se debe emplear una prueba no paramétrica. En relación a los resultados analizados en el indicador para determinar el tiempo promedio para la entrega del proyecto de software, se evidenció que el pre test en promedio es del 13.83%, mientras que en el post test se evidencia una reducción en promedio del 8.83%, lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, concluyendo que hay evidencias suficientes para plantear que la ficha de registro para determinar tiempo promedio para la entrega del proyecto de software es efectiva con un nivel de significación del 5%.

De acuerdo con Rudas (2017) en su estudio titulado: "Modelo de Gestión de riesgos para Proyectos de Desarrollo Tecnológico", en donde se evidencia en sus resultados los efectos positivos de implementar un modelo de gestión dentro de una empresa dedicada al desarrollo de proyectos, se traducen en ventajas palpables de reducción de costos, asegurando la calidad y reducción de tiempo del desarrollo de los proyectos. Dando como conclusión que la inclusión de un modelo de gestión en la estructura de la gestión de proyectos puede traducirse en beneficios actuales y futuros para obtener un aumento de resultados positivos en los objetivos de tiempo, costo y calidad. En la investigación de Carhuaricra (2018)

titulada “Sistema Web Para El Proceso De Control De Proyectos En La Empresa Gestión De Proyectos Informáticos & Sistemas”, se pudo conocer que al implementar una herramienta tecnológica permite al equipo de desarrollo incrementar los índices de rendimiento del cumplimiento del plan del trabajo y variación de los costos. Se concluye que hubo un aumento en el rendimiento del plan de trabajo, ya que el sistema permite planificar desde el inicio del procedimiento de control de los proyectos, reduciendo tiempo en el desarrollo de un proyecto, culminando los tiempos planificados según cronograma de trabajo.

Asimismo, guarda similitud con lo mencionado por Bayona et al. (2016) que señala que, como objetivo la gestión de proyectos canaliza de forma eficiente y productiva el trabajo de los desarrolladores, de tal forma que se conduce al éxito del proyecto, pues, dicho éxito incluye la gestión activa de los procesos con la intención de cumplir de manera exitosa con requisitos interesados, considerando el tiempo de desarrollo un elemento crucial al momento de medir la eficiencia y eficacia del desarrollo de un proyecto, por lo tanto, se debe tener en consideración en todo momento y plantear estrategias que permitan reducir tiempos entre procesos. Además, como menciona Ramos et al. (2017) gestionar un proceso de desarrollo de software mediante modelos, procesos, actividades o herramientas específicas, tiene sentido en el contexto de una empresa u organización, por lo que, es fundamental alinear las necesidades de la entidad basándose en el producto de software con la finalidad de gestionar actividades de desarrollo, control de tiempo, costos y calidad para la eficiencia del proyecto. En efecto, la gestión de proyectos es cada vez más importante para las organizaciones porque los proyectos son el método por el cual las organizaciones responden a su entorno. Siendo importante dentro de la gestión de proyectos son los estándares y métodos que se utilizan para controlar y llevar a cabo proyectos, conocidos colectivamente como métodos de gestión de proyectos.

Conforme al tercer objetivo específico de determinar el nivel de comunicación entre los involucrados del desarrollo del software de la Corte Superior de Justicia. Para comprobar la existencia de una diferencia significativa entre los resultados del pre test y post test en base a una ficha de registro, se empleó la prueba no paramétrica de Wilcoxon, en donde se concluyó que para medir la relación entre el Pre-Test y

el Post-Test se debe emplear una prueba no paramétrica. En relación a los resultados analizados en el indicador para determinar el nivel de comunicación con el área usuaria, se evidenció que el pre test en promedio es del 40.43%, mientras que en el post test se evidencia un aumento en promedio del 95.24%, lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, concluyendo que hay evidencias suficientes para plantear que la ficha de registro para determinar el nivel de comunicación con el área usuaria es efectiva con un nivel de significación del 5%.

De acuerdo con la investigación de Espinoza & Espinoza (2017) titulada "Marco de trabajo en base a design thinking y metodologías ágiles de desarrollo de software", en donde los resultados mostraron que con la implementación de metodologías ágiles en el desarrollo de software basado en design thinking, se logró aumentar de manera significativa el involucrar al usuario en el proyecto, lo cual permite identificar el involucramiento del usuario final en el proyecto elaborado, mejorando la comunicación entre todos los involucrados en el desarrollo del proyecto de software.

Además, los gerentes de proyecto deben enfocarse en comprender y atender las necesidades de comunicación de cada persona involucrada en el proyecto, lo cual implica determinar la comunicación e impacto en el transcurso del proyecto. Esto incluye actualizaciones de estado del proyecto, minutos de reuniones e informes sobre entregables, junto con otra documentación del proyecto (Ferrer, 2018). Lo mencionado indica que es importante la parte comunicativa, durante el tiempo de duración de ejecución de todo el proceso de desarrollo del software. Así mismo tiene relación con lo mencionado por Islam & Ferworn (2020), que indica que el método ágil requiere trabajo en equipo avanzado y habilidades relacionadas, puesto que, el flujo de trabajo se divide en unidades de trabajo encuadradas en el tiempo durante las cuales se va a realizar un subcomponente específico del proyecto entregado. Los entregables se priorizan de acuerdo con su valor comercial, es decir, esencialmente, según las especificaciones y necesidades del cliente. Si todo el trabajo planeado para una determinada unidad de trabajo no se puede completar, el trabajo se vuelve a priorizar y la información se utiliza para el futuro planificación de trabajo.

Conforme al cuarto objetivo específico de determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la Corte Superior de Justicia. Para comprobar la existencia de una diferencia significativa entre los resultados del pre test y post test en base a un cuestionario, se empleó la prueba no paramétrica de Wilcoxon, en donde se concluyó que para medir la relación entre el Pre-Test y el Post-Test se debe emplear una prueba no paramétrica. En relación a los resultados analizados en el indicador para determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia, se evidenció que el pre test en promedio es del 8.44%, mientras que en el post test se evidencia un aumento en promedio del 17.72%., lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, concluyendo que hay evidencias suficientes para plantear que el cuestionario para determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia es efectivo con un nivel de significación del 5%.

De acuerdo con la investigación de Maldonado & Cueva (2019) denominado "Gestión de proyectos de desarrollo de software orientado a modelos", en donde se concluye que un modelo de gestión de proyectos que se utiliza para controlar el desarrollo de software es importante, porque una metodología de gestión de proyectos eficaz se asocia con las ventajas del software orientado a modelos que brinda un modelo de calidad, lo que genera muchas ventajas en el proceso como en los niveles de satisfacción del usuario final.

Además, con lo mencionado por Mohammad et al. (2013) el desarrollo de software ágil o las metodologías ágiles son un grupo de procesos de desarrollo de software que se trata de realizar un desarrollo incremental o el desarrollo. Los requisitos y las soluciones del proceso de desarrollo ágil se cambian y evolucionan a través de colaboración y comunicación entre organización y equipos. Por ello, el desarrollo de un proyecto se basa en las personas involucradas con la finalidad de poder satisfacer las necesidades de los usuarios, siendo capaces de lidiar con cambios rápidos desde el lado del usuario o debido al entorno cambiante efectos (Pócsová et al., 2020).

VI. CONCLUSIONES

Tras la evaluación de los resultados se llegaron a las siguientes conclusiones:

- El modelo de gestión de proyectos aumenta el nivel de conocimiento sobre las metodologías ágiles de gestión de proyectos en la Corte Superior de Justicia, con aumento en promedio del 17.20%, lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, demostrando que hay evidencias suficientes para plantear que el cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles es efectivo con un nivel de significación del 5%.
- El modelo de gestión de proyectos disminuye el tiempo promedio de entrega de desarrollo de software, con una reducción en promedio del 8.83%., lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, demostrando que hay evidencias suficientes para plantear que la ficha de registro para determinar tiempo promedio para la entrega del proyecto de software es efectiva con un nivel de significación del 5%.
- El modelo de gestión de proyectos no aumenta el nivel de comunicación con el área usuaria durante el desarrollo del software, con un aumento en promedio del 95.24%, lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, demostrando que hay evidencias suficientes para plantear que la ficha de registro para determinar el nivel de comunicación con el área usuaria es efectiva con un nivel de significación del 5%.
- El modelo de gestión de proyectos aumenta nivel de satisfacción del usuario al finalizar el desarrollo, con un aumento en promedio del 17.72%, lo que indica que existe una diferencia significativa y se rechaza la hipótesis nula, demostrando que hay evidencias suficientes para plantear que el cuestionario para determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia es efectivo con un nivel de significación del 5%.

VII. RECOMENDACIONES

- Se recomienda mejorar las actividades del proceso de desarrollo de software con la implementación de un modelo de gestión de proyectos basado en metodología scrum.
- Se recomienda que en el área de desarrollo de software se capacite a los involucrados acerca del uso de las metodologías ágiles para mejorar los procesos de desarrollo.
- Para la implementación de un sistema de gestión de proyectos es recomendable considerar el tiempo de desarrollo como un factor clave.
- Se deben implementar estrategias para lograr un buen nivel de comunicación entre los involucrados del desarrollo de software.
- Para el área gerencial deben realizarse constantes estrategias de innovación para incrementar el nivel de satisfacción de los usuarios finales.

REFERENCIAS

- ABDALHAMID, S. y MISHRA, A., 2017. Adopting of Agile methods in Software Development Organizations: Systematic Mapping. *TEM Journal* [en línea], vol. 6, no. 4, pp. 817-825. [Consulta: 21 abril 2021]. DOI 10.18421/TEM64-22. Disponible en: <https://dx.doi.org/10.18421/TEM64-22>.
- ARIAS-GÓMEZ, J., VILLASÍS-KEEVER, M.Á. y MIRANDA-NOVALES, M.G., 2016. El protocolo de investigación III: la población de estudio. *Revista Alergia México* [en línea], vol. 63, no. 2, pp. 201-206. [Consulta: 21 abril 2021]. ISSN 2448-9190. DOI 10.29262/RAM.V63I2.181. Disponible en: <https://revistaalergia.mx/ojs/index.php/ram/article/view/181/309>.
- ARISPE ALBURQUEQUE, C.M., YANGALI VICENTE, J.S., GUERRERO BEJARANO, M.A., RIVERA LOZADA DE BONILLA, O., ACUÑA GAMBOA, L.A. y ARELLANO SACRAMENTO, C., 2020. PRESENTACIÓN DESARROLLO DEL INFORME FINAL DE TESIS CAPÍTULO I: EL PROBLEMA. [en línea], pp. 1-231. [Consulta: 24 abril 2021]. Disponible en: [https://repositorio.uide.edu.ec/bitstream/37000/4310/1/LA INVESTIGACIÓN CIENTÍFICA.pdf](https://repositorio.uide.edu.ec/bitstream/37000/4310/1/LA_INVESTIGACIÓN_CIENTÍFICA.pdf).
- ARIZA, D.A., 2017. Efectividad de la gestión de los proyectos: una perspectiva constructivista. *Obras y proyectos*, no. 22, pp. 75-85. ISSN 0718-2813. DOI 10.4067/s0718-28132017000200075.
- BAENA, G., 2017. *Metodología de la investigación*. 3. México: s.n.
- BARROS, A., 2016. *Proyectos TIC en el Sector Público*. ,
- BAYONA, J.C., PINEDA, O.L. y PARDO, O.D., 2016. El papel de la Ingeniería de Software en el desarrollo de aplicaciones. *Tecnología, Investigación y Academia* [en línea], vol. 4, no. 1, pp. 3-14. [28 abril 2021]. ISSN 2344-8288. Disponible en: <https://revistas.udistrital.edu.co/index.php/tia/article/view/5740>.
- CAMBRUZZI, W., RIGO, S. y BARBOSA, J., 2015. Dropout Prediction and Reduction in Distance Education Courses with the Learning Analytics Multitrail Approach. *Journal of Universal Computer Science*,

- CARHUARICRA HUAMÁN, A.A., 2018. Sistema web para el proceso de control de proyectos en la empresa gestión de proyectos Informáticos & Sistemas. *Universidad César Vallejo* [en línea], [21 mayo 2021]. Disponible en: <https://repositorio.ucv.edu.pe/handle/20.500.12692/19634>.
- CARRANZA GUERRERO, L.P., 2016. Gestión en Proyectos de Software. *Tecnología Investigación y Academia* [en línea], vol. 4, no. 2, pp. 12-19. [Consulta: 7 enero 2022]. Disponible en: <https://revistas.udistrital.edu.co/index.php/tia/article/view/7609>.
- CASTRO-RIVERA, V.P., HERRERA-ACUÑA, R.A. y VILLALOBOS-ABARCA, M.A., 2020. Desarrollo de un software web para la generación de planes de gestión de riesgos de software. *Información Tecnológica* [en línea], vol. 31, no. 3, pp. 135-148. [Consulta: 4 mayo 2021]. ISSN 07180764. DOI 10.4067/S0718-07642020000300135. Disponible en: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-07642020000300135&lng=es&nrm=iso&tlng=es.
- CÉSPEDES CAFFERATTA, A., 2017. Propuesta De Mejora Del Proceso De Gestión De Proyectos De Una Empresa Del Sector Tecnológico. , pp. 3.
- CUADROS, A.M., 2021. Fallos en gerencia de proyectos. *Universidad Pontificia Bolivariana.*
- DIMA, A.M. y MAASSEN, M.A., 2018. From Waterfall to Agile software: Development models in the IT sector. *Journal of International Studies*, vol. 11, no. 2, pp. 315-326. DOI 10.14254/2071.
- EMRAN, M., KABIR, H., RAHMAN, Z. y ISLAM, N., 2020. Enhancing Software Development Process Using Automated Adaptation of Object Ensembles. [en línea], [Consulta: 8 abril 2021]. Disponible en: <https://arxiv.org/ftp/arxiv/papers/2005/2005.03241.pdf>.
- ESPINOZA, J. y ESPINOZA, E., 2017. Marco de trabajo en base a Design Thinking y metodologías ágiles de desarrollo de software. *Universidad Peruana de Ciencias Aplicadas* [en línea]. [Consulta: 7 abril 2021]. Disponible en: <https://arxiv.org/ftp/arxiv/phttps://repositorioacademico.upc.edu.pe/bitstre>

am/handle/10757/622507/Espinoza_VJ.pdf?sequence=5&isAllowed=yapers/2005/2005.03241.pdf.

- FERNÁNDEZ, E., HOLANDA, M., VICTORINO, M., BORGES, V., CARVALHO, R. y ERVEN, G., 2019. Educational data mining: Predictive analysis of academic performance of public school students in the capital of Brazil. *Journal of Business Research*,
- FERRER ROMERO, E.F., 2018. Strategic project management: a methodology for sustainable competitive advantage. *Rev. esc.adm.neg. Edición Especial* [en línea], pp. 15-31. [Consulta: 5 mayo 2021]. DOI 10.21158/01208160.n0.2018.2016. Disponible en: <https://doi.org/10.21158/01208160.n0.2018.2016>.
- HALVORSEN, H., 2020. *Software Development A Practical Approach. Publisher,*
- HERNÁNDEZ, R., FERNÁNDEZ, C. y BAPTISTA, M., 2014. Metodología de la Investigación. *México D.F*, vol. 5.
- HERNÁNDEZ, R. y MENDOZA, C., 2018. Metodología de la investigación. Las rutas cuantitativa, cualitativa y mixta. *Editorial Mc Graw Hill Education*,
- HOSSEN, A., 2016. An inventory model with price and time dependent demand with fuzzy valued inventory costs under inflation. *Pure Appl*, vol. 11, no. 2.
- HRON, M. y OBWEGESER, N. 2021. Why and how is Scrum being adapted in practice: A systematic review. *Journal of Systems and Software*. Disponible en: <https://www.scopus.com/record/display.uri?eid=2-s2.0-85117250893&origin=resultslist&sort=plf-f&src=s&st1=Scrum+methodology&sid=ca7931d2da8b3102a3a4c32b71c4882b&sot=b&sdt=b&sl=32&s=TITLE-ABS-KEY%28Scrum+methodology%29&relpos=3&citeCnt=0&searchTerm=>
- ILLICACHI, A. y OLMEDO, F., 2020. Propuesta de mejora para el proceso de desarrollo de software del Consejo Nacional Electoral. [en línea], [Consulta: 14 mayo 2021]. Disponible en: https://repositorio.epneumann.edu.pe/bitstream/handle/20.500.12892/247/TRABAJO_DE_INV_OLMEDO_FAUSTO-

ILLICACHI_ANGELICA.pdf?sequence=1&isAllowed=y.

- INEI, 2017. Instituto Nacional de Estadística e Informática. [en línea]. [Consulta: 7 enero 2022]. Disponible en: <http://www.inei.gob.pe/biblionei/bancopub/Est/Lib0038/N10/cuadc022-027>.
- ISLAM, A.K.. Z. y FERWORN, D.A., 2020. A Comparison between Agile and Traditional Software Development Methodologies. *Global Journal of Computer Science and Technology* [en línea], pp. 7-42. [Consulta: 4 mayo 2021]. ISSN 09754350. DOI 10.34257/gjcstcvol20is2pg7. Disponible en: https://globaljournals.org/GJCST_Volume20/2-A-Comparison-between-Agile.pdf.
- JINJIN, L. y BERLIN, G., 2020. Agile Software Development. *Universitt berlin* [en línea], [Consulta: 4 mayo 2021]. Disponible en: <https://www.snet.tu-berlin.de/fileadmin/fg220/courses/WS1112/snet-project/agile-software->.
- KYOO, S., 2016. Innovative Software Development and Project Management Framework for Technology Startups. *Arxiv* [en línea], Disponible en: <https://arxiv.org/pdf/1708.06900>.
- MACAPAGAL, M., 2010. ICT project management in theory and practice. *Apcict* [en línea], no. 7. Disponible en: https://www.unapcict.org/sites/default/files/2018-12/BriefingNote-7-web_0.pdf.
- MALDONADO, T. y CUEVA, S., 2019. Gestión de proyectos de desarrollo de software orientado a modelos. [en línea], Disponible en: https://www.researchgate.net/publication/333659086_Gestion_de_proyectos_de_desarrollo_de_software_orientado_a_modelos.
- MIGHETTI, J. y HADAD, G., 2016. A Requirements Engineering Process Adapted to Global Software Development. *Clei Electronic Journal*, [en línea], vol. 19, no. 3. Disponible en: <http://www.scielo.edu.uy/pdf/cleiej/v19n3/0717-5000-cleiej-19-03-00181.pdf>.
- MIRANDA, M. y GUZMÁN, J., 2017. Análisis de la Deserción de Estudiantes

- Universitarios usando Técnicas de Minería de Datos. *Formación universitaria*, vol. 10, no. 3.
- MOHAMMED, O., 2016. Hybrid energy storage management in ship power systems with multiple pulsed loads. *Electric Power Systems Research*, vol. 141.
- MOLINA, B., VITE, H. y DÁVILA, J., 2018. Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *Espirales revista multidisciplinaria de investigación* [en línea], Disponible en: <http://revistaespirales.com/index.php/es/article/view/269/225>.
- MOURA, R.L. de, CARNEIRO, T.C.J. y DINIZ, B.D., 2018. Influence of the project manager's personal characteristics on project performance. *Gestao e Producao* [en línea], vol. 25, no. 4, pp. 751-763. [Consulta: 4 mayo 2021]. ISSN 18069649. DOI 10.1590/0104-530X3595-16. Disponible en: <https://doi.org/10.1590/0104-530X3595-16>.
- NIETO, E., 2018. Tipos de Investigación. [en línea], Disponible en: <http://repositorio.usdg.edu.pe/handle/USDG/34>.
- OTZEN, T. y MANTERLA, C., 2017. Técnicas de Muestreo sobre una Población a Estudio. [en línea], Disponible en: <https://scielo.conicyt.cl/pdf/ijmorphol/v35n1/art37.pdf>.
- PÉREZ, A., 2021. Metodología de gestión de proyectos: cuatro modelos. [en línea], Disponible en: <https://www.obsbusiness.school/blog/metodologia-de-gestion-de-proyectos-cuatro-modelos>.
- PÓCSOVÁ, J., BEDNÁROVÁ, D., BOGDANOVSKÁ, G. y MOJŽIŠOVÁ, A., 2020. Implementation of Agile Methodologies in an Engineering Course. *Education Sciences* [en línea], Disponible en: <https://www.mdpi.com/2227-7102/10/11/333/pdf>.
- PROJECT-SUCCESS-QRC-STANDISH-GROUP-CHAOS-REPORT., 2020. Project Success Quick Reference Card. ,
- RADA CLAVIJO, M.R., 2013. LA GESTIÓN DE PROYECTOS TECNOLÓGICOS. *Fides et Ratio*,

- RAMÍREZ, P. y GRANDÓN, E., 2018. Prediction of Student Dropout in a Chilean Public University through Classification based on Decision Trees with Optimized Parameters. *Revista SciELO* [en línea], vol. 11, no. 3. Disponible en: https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-50062018000300003&lng=pt&nrm=i.p&tlng=es ISSN: 0718-5006.
- RAMOS, D., NORIEGA, R., LAÍNEZ, J. y DURANGO, A., 2017. Curso de Ingeniería de Software. *IT Campus Academy*.
- RODRÍGUEZ, R., 2016. Propuesta de un modelo de gestión de proyectos TI, para desarrollar software, basado en las metodologías y buenas prácticas de la industria. Caso de estudio BusinessIT. *Escuela Politécnica Nacional* [en línea], Disponible en: <https://bibdigital.epn.edu.ec/handle/15000/16948>.
- RUDAS, L., 2017. Modelo de gestión de riesgos para proyectos de desarrollo tecnológico. [en línea], Disponible en: <https://ciateq.repositorioinstitucional.mx/jspui/bitstream/1020/86/1/RudasTayoLeidyPMDGPI2017.pdf>.
- RUVALCABA, M., 2020. Procesos de Software. [en línea], Disponible en: <https://sg.com.mx/revista/1/procesos-software>.
- SAEEDI, K. y VISVIZI, A., 2021. Software Development Methodologies, HEIs, and the Digital Economy. *Education Sciences* [en línea], vol. 11, no. 73. Disponible en: <https://files.eric.ed.gov/fulltext/EJ1288479.pdf>.
- SAWALHA, S. y NABI, H., 2020. Agile Software Development: Methodologies and Trends. [en línea], vol. 14, no. 11. Disponible en: https://www.researchgate.net/publication/342848746_Agile_Software_Development_Methodologies_and_Trends.
- SIEGEL, E., 2013. Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die. *Wiley*, vol. 10.
- SIFUENTES, O., 2018. Predictive models of student desertion at a private Peruvian university. *Revista Producción y Gestión. Revista Producción y Gestión* [en línea], vol. 21, no. 2. Disponible en:

<https://www.redalyc.org/jatsRepo/816/81658967008/html/index.html> .

SUBIRIA, J. y RAMÍREZ, A., 2019. Cómo Investigar en Educación. *Cooperativa Editorial “Magisterio”*, no. 1.

UNIVERSIDAD ESAN, 2019. ConsexionEsan. [en línea], Disponible en: <https://www.esan.edu.pe/apuntes-empresariales/2019/09/gestion-de-proyectos-cuales-son-los-problemas-mas-usuales/>.

VIALE TUDELA, H., 2014. UNA APROXIMACIÓN TEÓRICA A LA DESERCIÓN ESTUDIANTIL UNIVERSITARIA. *Revista Digital de Investigación en Docencia Universitaria*, vol. 1.

VINUEZA, C., 2021. Diseño de un modelo matemático para estimar la deserción estudiantil mediante técnicas de análisis multivariado en una institución de educación superior tecnológica. *Universidad Técnica de Ambato* [en línea], Disponible en: <https://repositorio.uta.edu.ec/bitstream/123456789/32219/1/t1765mma.pdf>.

YUNIA, J. y URBANO, C., 2018. Mapas y Herramientas para conocer la escuela: Investigación Etnográfica e Investigación – Acción. , no. 3.

ANEXOS

Anexo 1. Matriz de operacionalización de las variables

VARIABLES DE ESTUDIO	DEFINICIÓN CONCEPTUAL	DEFINICIÓN OPERATIVA	DIMENSIONES	INDICADORES	ESCALA DE MEDICIÓN
Modelo de gestión de proyectos	El proceso de software es el objetivo de la ingeniería de software. Se trata de un conjunto de personas, estructura organizacional, reglas, políticas, actividades y procedimientos, componentes de software, metodología y herramientas que se utilizan específicamente para definir, desarrollar, brindar servicios, innovar y expandir un producto de software. (Ruvalcaba, 2020),	El modelo de gestión de proyectos seguirá la metodología ágil, en el marco de trabajo scrum ya que está destinada principalmente al desarrollo y mantenimiento de proyectos de software.	Planificación	Nivel de viabilidad	De Razón
			Desarrollo	Nivel de flujo de trabajo	
			Cierre	Nivel de eficacia	
	Un proceso de desarrollo del software es el proceso de dividir	El proceso de desarrollo de software		Nivel de conocimiento de los equipos de	De Razón

Proceso de Desarrollo de Software	el trabajo de desarrollo del software en distintas fases para mejorar el diseño, la gestión del producto, y la gestión de proyecto. Es también conocido como el ciclo de vida del desarrollo de software. (Ramírez y Grandon, 2018)	se mide a través de las dimensiones de los tiempos que se emplean dentro de la Corte Superior de Justicia de Lambayeque	Curva de Aprendizaje	desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.
			Tiempo de desarrollo	Tiempo promedio de entrega de desarrollo de software.
			Comunicación con el usuario	Nivel de comunicación con el área usuaria durante el desarrollo del software.
			Aceptación del usuario final	Nivel de satisfacción del usuario

Anexo 2. Cuadro de Indicadores

OBJETIVOS ESPECÍFICOS	INDICADORES	DESCRIPCIÓN	TÉCNICA O INSTRUMENTO	FÓRMULA
Aumentar el nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.	Nivel de conocimiento sobre metodologías de gestión de proyectos	Determinar el nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.	Cuestionario	$\text{NCMG} = \frac{\sum_{i=1}^n (\text{SNO})i}{n}$ <p>NCMG= Nivel de conocimiento sobre metodologías de gestión de proyectos.</p> <p>SNO = Sumatoria de la nota obtenida</p> <p>n = Número de trabajadores que fueron encuestados</p>
Reducir el tiempo promedio para la entrega del software con el apoyo de una plataforma informática que permita planificar los requerimientos de la Corte Superior de Justicia	Tiempo promedio para la entrega del proyecto de software	Determinar el tiempo promedio para la entrega del proyecto de software	Ficha de registro	$\text{TPEDS} = \frac{TDEP}{TPEP} \times 100$ <p>TPEDS= Tiempo promedio de entrega de desarrollo de software.</p> <p>TDEP = Tiempo de demora para la entrega del producto</p> <p>TPEP= Tiempo planificado para la entrega del producto</p>

<p>Aumentar el nivel de comunicación entre los involucrados del desarrollo del software de la Corte Superior de Justicia</p>	<p>Nivel de comunicación entre los involucrados del desarrollo del software</p>	<p>Determinar el nivel de comunicación con los involucrados en el desarrollo del software.</p>	<p>Ficha de Registro</p>	<p>$NCAU = \frac{NRE}{NRP} \times 100$</p> <p>NCAU= Nivel de comunicación con el área usuaria.</p> <p>NRE= Número de reuniones ejecutadas.</p> <p>NRP= Número de reuniones programadas.</p>
<p>Aumentar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la Corte Superior de Justicia.</p>	<p>Nivel de Satisfacción del usuario al finalizar el desarrollo del software</p>	<p>Determinar el nivel de satisfacción del usuario al finalizar el desarrollo del software de la Corte Superior de Justicia.</p>	<p>Cuestionario</p>	<p>$NSUFDS = \frac{CO}{CMT} \times 100$</p> <p>NSUFDS = Nivel de Satisfacción del usuario al finalizar el desarrollo del software.</p> <p>CO = Calificación Obtenida</p> <p>CTM = Calificación máxima total</p>

Anexo 3. Instrumentos de validación de datos

Anexo 3.1 Instrumento para identificar problemática

Entrevista al coordinador de informática para conocer la realidad problemática de la Corte Superior de Justicia

- 1) ¿Cuántos proyectos se han ejecutado en la Corte Superior de Justicia de Lambayeque?
- 2) ¿Tienen proyectos que no han terminado de cumplir con el objetivo propuesto?
- 3) ¿Cuenta con alguna herramienta informática que permita realizar el seguimiento del desarrollo de los proyectos informáticos?
- 4) ¿Cuáles son las actividades que realizan al momento de iniciar un proyecto informático?
- 5) ¿Qué medio utilizan para realizar la comunicación con el área usuaria?
- 6) ¿Qué tipo de documentación elaboran para el desarrollo del proyecto?
- 7) ¿Qué debilidades se les ha presentado para poder lograr que el proyecto sea exitoso?
- 8) ¿Han logrado cumplir con el cronograma de los proyectos y existe una planificación para este?

Anexo 3.2. Cuestionario para determinar el nivel de conocimiento sobre metodologías ágiles.

CUESTIONARIO PARA DETERMINAR EL NIVEL DE CONOCIMIENTO SOBRE METODOLOGÍAS ÁGILES.			
Investigador		Tipo de prueba	
Empresa en estudio			
Método de investigación			
Fecha inicio		Fecha de fin	

Objetivo	Indicador	Medida	Fórmula
Aumentar el nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.	Nivel de conocimiento sobre metodologías de gestión de proyectos	Porcentaje	$NCMG = \frac{\sum_{i=1}^n (SNO)i}{n}$ <p>NCMG= Nivel de conocimiento sobre metodologías de gestión de proyectos.</p> <p>SNO = Sumatoria de la nota obtenida</p> <p>n = Número de trabajadores que fueron encuestados</p>

N°	Interrogante	SI	NO
METODOLOGÍA ÁGIL			
01	Los equipos en un proyecto ágil se caracterizan por ser autoorganizados y colaborativos.	SI	NO
02	En un proyecto ágil cada iteración concluye con un prototipo funcional.	SI	NO
03	Según los doce principios del manifiesto ágil, el método más eficiente y efectivo para hacer llegar la información a o dentro de un equipo de desarrollo es EL DIÁLOGO CARA A CARA.	SI	NO
04	¿Considera necesario no documentar, ya que no es parte del segundo principio agile?	SI	NO
05	En el manifiesto por el desarrollo ágil de software se valora más el seguimiento estricto de un plan sobre responder a los cambios	SI	NO
06	¿En un equipo ágil, el propietario del producto es el representante del cliente?	SI	NO

07	Una de las metodologías más comunes en el desarrollo tanto de empresas grandes como de empresas pequeñas es SCRUM.	SI	NO
SCRUM			
08	¿Scrum y Agile es lo mismo??	SI	NO
09	¿Scrum, se ha diseñado y pensado, únicamente para el sector tecnológico?	SI	NO
10	El Scrum Master, realiza todas las tareas de un Project Manager	SI	NO
11	¿El Scrum Master es un líder que está al servicio del Equipo Scrum?	SI	NO
12	¿Es cierto que el Product Owner es el encargado de gestionar el progreso del trabajo durante el Sprint?	SI	NO
13	¿Considera importante que el equipo de desarrollo realice reuniones sin la participación del Product Owner (Dueño del Producto)?	SI	NO
14	¿Cada miembro del equipo es completamente responsable de su tarea (estimación, codificación, pruebas)?	SI	NO
15	En el Sprint Backlog es el Product Backlog dividido en entregas	SI	NO
16	En la Revisión de Sprint el equipo scrum y los interesados conversan cómo se puede optimizar el valor del producto y cómo aprovechar las nuevas oportunidades de negocio.	SI	NO

Anexo 3.3. Ficha de registro para determinar tiempo promedio para la entrega del proyecto de software.

FICHA DE REGISTRO PARA DETERMINAR TIEMPO PROMEDIO PARA LA ENTREGA DEL PROYECTO DE SOFTWARE.			
Investigador		Tipo de prueba	
Empresa en estudio			
Método de investigación			
Fecha inicio		Fecha de fin	

Objetivo	Indicador	Medida	Fórmula
Reducir el tiempo promedio para la entrega del software con el apoyo de una plataforma informática que permita planificar los requerimientos de la Corte Superior de Justicia	Tiempo promedio para la entrega del proyecto de software	Porcentaje	$TPEDS = \frac{TDEP}{TPEP} \times 100$ <p>TPEDS= Tiempo promedio de entrega de desarrollo de software.</p> <p>TDEP = Tiempo de demora para la entrega del producto</p> <p>TPEP= Tiempo planificado para la entrega del producto</p>

PRE TEST							
Nombre del Proyecto							
Descripción	Tiempo planificado	Días De Ampliación	Tiempo Aplazado	Tiempo Entregado	Demora	% Tiempo de Entrega	Observaciones
Entregable 1							
Entregable 2							
Entregable 3							

Entrega ble 4							
Produc to Final							
Implem entació n							
POS TEST							
Nombre del Proyecto							
Descrip ción	Tiempo planificado	Días De Ampliación	Tiempo Aplazado	Tiempo Entregado	Demora	% Tiempo de Entrega	Observac iones
Entrega ble 1							
Entrega ble 2							
Entrega ble 3							
Entrega ble 4							
Produc to Final							
Implem entació n							

Anexo 3.4. Ficha de registro para determinar el nivel de comunicación con el área usuaria

FICHA DE REGISTRO PARA DETERMINAR EL NIVEL DE COMUNICACIÓN CON EL ÁREA USUARIA			
Investigador		Tipo de prueba	
Empresa en estudio			
Método de investigación			
Fecha inicio		Fecha de fin	

Objetivo	Indicador	Medida	Fórmula
Aumentar el nivel de comunicación entre los involucrados del desarrollo del software de la Corte Superior de Justicia	Nivel de comunicación entre los involucrados del desarrollo del software	Porcentaje	$NCAU = \frac{NRE}{NRP} \times 100$ <p>NCAU= Nivel de comunicación con el área usuaria. NRE= Número de reuniones ejecutadas. NRP= Número de reuniones programadas.</p>

PRE TEST						
Nombre del Proyecto			Responsable	Reuniones programadas	Reuniones ejecutadas	% de nivel de comunicación
Semana	Fecha	Descripción				
Semana 01						
Semana 02						
Semana 03						
Semana 04						
Semana 05						
Semana 06						

Semana 07						
POS TEST						
Nombre del Proyecto						
Semana	Fecha	Descripción	Responsable	Reuniones programadas	Reuniones ejecutadas	% de nivel de comunicación
Semana 01						
Semana 02						
Semana 03						
Semana 04						
Semana 05						
Semana 06						
Semana 07						

Anexo 3.4. Cuestionario para determinar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la corte superior de justicia.



CUESTIONARIO PARA DETERMINAR EL NIVEL DE SATISFACCIÓN DE LOS USUARIOS AL FINALIZAR EL DESARROLLO DEL SOFTWARE DE LA CORTE SUPERIOR DE JUSTICIA.			
Investigador		Tipo de prueba	
Empresa en estudio			
Método de investigación			
Fecha inicio		Fecha de fin	

Objetivo	Indicador	Medida	Fórmula
Aumentar el nivel de satisfacción de los usuarios al finalizar el desarrollo del software de la Corte Superior de Justicia.	Nivel de Satisfacción del usuario al finalizar el desarrollo del software	Porcentaje	$NSUFDS = \frac{CO}{CMT} \times 100$ <p>NSUFDS= Nivel de Satisfacción del usuario al finalizar el desarrollo del software.</p> <p>CO = Calificación Obtenida</p> <p>CTM = Calificación máxima total</p>

Corte Superior de Justicia de Lambayeque			
EXCELENTE	BUENO	REGULAR	PÉSIMO
4	3	2	1
CARACTERÍSTICAS	PUNTAJE	CRITERIOS	CALIFICACIÓN
CUMPLIMIENTO Y ENTREGA	4	EXCELENTE. - El servicio o producto se entregó en la fecha pactada.	4
	3	BUENO. - El servicio o producto se entregó en la fecha, pero con observaciones.	
	2	REGULAR. - El servicio o producto se entregó posterior a la fecha pactada, pero no superior a 2 días.	
	1	PÉSIMO. - El servicio o producto se entregó en fecha posterior a lo pactado, superior a 7 días.	
CALIDAD Y ALCANCE DEL PROYECTO	4	EXCELENTE. - Supera las expectativas y mejora el alcance establecido para el bien y/o servicio entregado.	4
	3	BUENO. - Cumplió con los requisitos y alcance establecido para el bien y/o servicio adquirido.	
	2	REGULAR. - Faltó a uno o más requisitos y/o especificaciones técnicas, que previo requerimiento fueron subsanadas sin incurrir en perjuicios para el cliente.	

	1	PÉSIMO. - Le presentó inconformidades graves en la calidad y cumplimiento del alcance, ocasionando incumplimiento del contrato y dando lugar a la aplicación de garantías.	
PROCESO DEL DESARROLLO DEL SOFTWARE	4	EXCELENTE. - El equipo de desarrollo ha cumplido con sus expectativas en las reuniones de planificación y revisión de entregables, entregando mejoras en cada iteración.	4
	3	BUENO. - El equipo de desarrollo ha cumplido con las reuniones de planificación y revisión de entregables.	
	2	REGULAR. - El equipo de desarrollo no ha tenido una comunicación eficiente con el área durante las reuniones de planificación y revisión de entregables, dando falsas expectativas.	
	1	PÉSIMO:- El equipo de desarrollo no se reunió con el área usuaria tan solo en las reuniones de revisión, no cumpliendo con el alcance de cada entregable.	
PRODUCTO FINAL	4	EXCELENTE: El producto es útil ayudando a incrementar la efectividad en la operación de los procesos de la corte superior de justicia	4
	3	BUENO. - El producto final ha sido finalizado y ayudará con la sistematización de los procesos.	
	2	REGULAR. - El producto final ha sido elaborado, pero no todos los módulos están definidos correctamente según las reglas de negocio.	
	1	PÉSIMO: El producto no ayuda a incrementar la efectividad en la operación de los procesos de la corte superior de justicia	
CAPACITACIONES Y DOCUMENTACIÓN	4	EXCELENTE: El equipo de desarrollo realizó capacitaciones al área usuaria sobre el funcionamiento del sistema entregando documentación de ayuda como manuales de usuarios, cartillas entre otros documentos.	4
	3	BUENO: El equipo de desarrollo realizó una capacitación y entregó documentación de ayuda como un manual de usuario.	
	2	REGULAR: El equipo de desarrollo no entregó documentación de ayuda para el área.	
	1	PÉSIMO: El equipo de desarrollo no realizó capacitaciones ni documentación del sistema.	
TOTAL			20.0
% GENERAL DE ACEPTACIÓN			100%
GRADO DE SATISFACCIÓN DEL CLIENTE			BUENO
GRADO DE SATISFACCIÓN DEL CLIENTE			
[0% - 50%]	[51 % - 79%]		[80% - 100%]
MALO	REGULAR		BUENO

Anexo 4. Matriz de evaluación de expertos

 MATRIZ DE EVALUACIÓN DE EXPERTOS				
Título de la investigación:	"Modelo de Gestión de Proyectos para mejorar el Proceso de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque, 2021"			
Línea de investigación:	SISTEMAS DE INFORMACIÓN			
El instrumento de medición pertenece a las variables:	VI: INTERFAZ DE INTEGRACIÓN VD: TOMA DE DECISIONES			
<p>Mediante la matriz de evaluación de expertos. Ud. Tiene la facultad de evaluar cada una de las preguntas marcando con una "x" en las columnas de SÍ o NO. Asimismo, le exhortamos en la corrección de los ítems indicando sus observaciones con la finalidad de mejorar la coherencia de las preguntas sobre la variable en estudio</p>				
Ítems	Preguntas	Aprecia		Observaciones
		SÍ	NO	
1	¿El instrumento de medición presenta el diseño adecuado?	X		
2	¿El instrumento de recolección de datos tiene relación con el título de la investigación?	X		
3	¿El instrumento de recolección de datos se mencionan las variables de investigación?	X		
4	¿El instrumento de recolección de datos facilitará el logro de los objetivos de la investigación?	X		
5	¿El instrumento de recolección de datos se relaciona con las variables de estudio?	X		
6	¿La redacción de las preguntas tienen un sentido coherente y no están sesgadas?	X		
7	¿Cada una de las preguntas del instrumento de medición se relaciona con cada uno de los elementos de los indicadores?	X		
8	¿El diseño del instrumento de medición facilitará el análisis y procesamiento de datos?	X		
9	¿Son entendibles las alternativas de respuesta del instrumento de medición?	X		
10	¿El instrumento de medición será accesible a la población sujeto de estudio?	X		
11	¿El instrumento de medición es claro, preciso y sencillo de responder para, de esta manera, obtener los datos requeridos?	X		
<p>Sugerencias:</p> <p style="text-align: center;">Aplicable</p>				
<p>Nombre completo: Toño Alvites Adan DNI: 48017627 Grado. Maestro en Gestión Pública</p>		 Toño Alvites Adán INGENIERO INDUSTRIAL REG. CIP. 213206 <hr/> Firma del Experto		



MATRIZ DE EVALUACIÓN DE EXPERTOS

Título de la investigación:	"Modelo de Gestión de Proyectos para mejorar el Proceso de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque, 2021"
Línea de investigación:	SISTEMAS DE INFORMACIÓN
El instrumento de medición pertenece a las variables:	VI: INTERFAZ DE INTEGRACIÓN VD: TOMA DE DECISIONES

Mediante la matriz de evaluación de expertos. Ud. Tiene la facultad de evaluar cada una de las preguntas marcando con una "x" en las columnas de SÍ o NO. Asimismo, le exhortamos en la corrección de los ítems indicando sus observaciones con la finalidad de mejorar la coherencia de las preguntas sobre la variable en estudio

Ítems	Preguntas	Aprecia		Observaciones
		SÍ	NO	
1	¿El instrumento de medición presenta el diseño adecuado?	X		
2	¿El instrumento de recolección de datos tiene relación con el título de la investigación?	X		
3	¿El instrumento de recolección de datos se mencionan las variables de investigación?	X		
4	¿El instrumento de recolección de datos facilitará el logro de los objetivos de la investigación?	X		
5	¿El instrumento de recolección de datos se relaciona con las variables de estudio?	X		
6	¿La redacción de las preguntas tienen un sentido coherente y no están sesgadas?	X		
7	¿Cada una de las preguntas del instrumento de medición se relaciona con cada uno de los elementos de los indicadores?	X		
8	¿El diseño del instrumento de medición facilitará el análisis y procesamiento de datos?	X		
9	¿Son entendibles las alternativas de respuesta del instrumento de medición?	X		
10	¿El instrumento de medición será accesible a la población sujeto de estudio?	X		
11	¿El instrumento de medición es claro, preciso y sencillo de responder para, de esta manera, obtener los datos requeridos?	X		

Sugerencias:

Aplicable

Nombre completo: Shirley Serquén Bravo
DNI: 45498435
Grado. Maestro en Administración de Empresas

Firma del Experto



MATRIZ DE EVALUACIÓN DE EXPERTOS

Título de la investigación:	"Modelo de Gestión de Proyectos para mejorar el Proceso de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque, 2021"
Línea de investigación:	SISTEMAS DE INFORMACIÓN
El instrumento de medición pertenece a las variables:	VI: INTERFAZ DE INTEGRACIÓN VD: TOMA DE DECISIONES

Mediante la matriz de evaluación de expertos. Ud. Tiene la facultad de evaluar cada una de las preguntas marcando con una "x" en las columnas de SÍ o NO. Asimismo, le exhortamos en la corrección de los ítems indicando sus observaciones con la finalidad de mejorar la coherencia de las preguntas sobre la variable en estudio

Ítems	Preguntas	Aprecia		Observaciones
		SÍ	NO	
1	¿El instrumento de medición presenta el diseño adecuado?	X		
2	¿El instrumento de recolección de datos tiene relación con el título de la investigación?	X		
3	¿El instrumento de recolección de datos se mencionan las variables de investigación?	X		
4	¿El instrumento de recolección de datos facilitará el logro de los objetivos de la investigación?	X		
5	¿El instrumento de recolección de datos se relaciona con las variables de estudio?	X		
6	¿La redacción de las preguntas tienen un sentido coherente y no están sesgadas?	X		
7	¿Cada una de las preguntas del instrumento de medición se relaciona con cada uno de los elementos de los indicadores?	X		
8	¿El diseño del instrumento de medición facilitará el análisis y procesamiento de datos?	X		
9	¿Son entendibles las alternativas de respuesta del instrumento de medición?	X		
10	¿El instrumento de medición será accesible a la población sujeto de estudio?	X		
11	¿El instrumento de medición es claro, preciso y sencillo de responder para, de esta manera, obtener los datos requeridos?	X		

Sugerencias:

Aplicable

Nombre completo: Hugo Hamilton Oyola Yarlaque
DNI: 70409180
Grado. Maestro en Administración de Empresas


Firma del Experto

Anexo 5. Carta de aceptación de la empresa



"Año del Bicentenario del Perú: 200 de años de Independencia"
CORTE SUPERIOR DE JUSTICIA DE LAMBAYEQUE

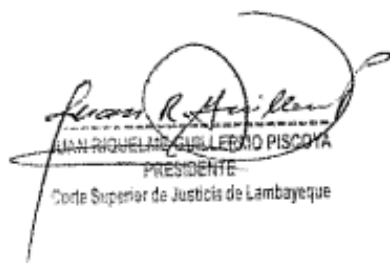
Presidencia

Referencia: Carta N° 096-2021-UCV-VA-EPG-SL01/J.

Asunto: Solicitud de Autorización para realizar investigación.

Chiclayo, 18 de noviembre de dos mil veintiuno.-

DADO CUENTA con el documento de la referencia, cursado por el Dr. Emilio Alberto Soriano Paredes, en calidad de Jefe de la Escuela de Posgrado de la Universidad César Vallejo Filial Trujillo; mediante el cual solicita autorización para que la señora Jessica Katherine Carrasco Zeña, estudiante de maestría en Ingeniería de Sistemas con mención en Tecnologías de la Información realice su trabajo de investigación denominado "Modelo de Gestión de Proyectos para mejorar el Proceso de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque". Asimismo, a través de una carta, la señorita en mención, aclara que los instrumentos que se van aplicar son cuestionarios al personal del área de informática, más no se realizará una manipulación al software o sistema informático de esta Corte; estando a lo expuesto: i) **AUTORIZAR** a la señora Jessica Katherine Carrasco Zeña y solo con fines académicos, acceder a la información para el desarrollo de su investigación, previa coordinación con el Coordinador de Informática de esta Corte. ii) **PONER DE CONOCIMIENTO** al Coordinador de Informática de la Corte Superior de Justicia de Lambayeque, a fin de que conforme a sus atribuciones y en lo que corresponda, atienda lo solicitado, dando cuenta a esta Presidencia de lo accionado, sin afectar las actividades laborales propias a su función. iii) **NOTIFIQUESE**.-



JUAN RIQUELME GULLETO PISCOTZA
PRESIDENTE
Corte Superior de Justicia de Lambayeque

Anexo 6. Carta de conformidad de la empresa



"Año del Bicentenario del Perú: 200 años de Independencia"
CORTE SUPERIOR DE JUSTICIA DE LAMBAYEQUE

Chiclayo, 28 de diciembre del 2021

Referencia: Carta N°096-2021-UCV-VA-EPG-SL01/J.

Dr. Emilio Alberto Soriano Paredes
Jefe de la Escuela de Posgrado – Trujillo
Universidad César Vallejo
Asunto: Conformidad del Proyecto de Investigación

Es grato dirigirme a usted para saludarlo cordialmente y al mismo tiempo hacer de su conocimiento que, en mérito de lo solicitado por la Br. Jessica Katherin Carrasco Zeña identificada con DNI: 72470890 ha entregado resultados al área de informática correspondiente a su trabajo de investigación titulado "Modelo de Gestión de Proyectos para mejorar el Proceso de Desarrollo de Software de la Corte Superior de Justicia de Lambayeque, 2021" el cual servirá de gran ayuda para los futuros proyectos de software de nuestra Institución. Por lo tanto, otorgo mi conformidad de su implementación.

Sin otro particular, hago llegar a usted las muestras de mi consideración y estima.

Atentamente,

MARCO ANTONIO CESPEDES BURGA
JEFE DEL ÁREA DE INFORMÁTICA
Corte Superior de Justicia de Lambayeque

Sede: "Manuel Lorenzo Emerson Huangal Noveda" – Av. José Leonardo Ortiz N°155 C.C. Chiclayo
Teléfono 074-481640 Anexo 22324- presidencia@pj.gob.pe

Anexo 7. Modelo de gestión de proyectos

Anexo 7.1. Nivel de conocimiento de los equipos de desarrollo sobre las metodologías de gestión de proyectos en la Corte Superior de Justicia.

Procesos de desarrollo de software

Marco de trabajo Scrum

Fases:

Fase de Inicio (sprint 0)

Fase de planificación (sprint 1)

Fase de implementación (sprints)

Fase de Cierre

1. Introducción

En la investigación presentada se tomó como elección el desarrollo de sus procesos y desarrollo de software bajo el marco metodológico SCRUM, así mismo dentro del presente escrito encontrará funciones y responsabilidad de los colaboradores, así como de las partes interesadas (áreas usuarias) en el proyecto

2. Desarrollo de la Metodología

2.1 Fundamentación:

La metodología empleada en la siguiente investigación se basa en los procesos del marco de trabajo Scrum, donde se le propone a la Corte Superior de Justicia implementar actividades con un flujo de proyectos Scrum para el desarrollo de sus Proyectos de Software. Estos procesos se agrupan en las siguientes fases como se muestra en la Tabla 01 Inicio, Planificación y estimación, Implementación y Cierre acompañadas de ciertas actividades. Esta propuesta se ha realizado con apoyo del *Workbook SCD Scrum Study- Targeting*.

Tabla 25. Resumen de los Procesos de Scrum

Fase	Procesos fundamentales de scrum
I. Inicio	<ol style="list-style-type: none">1. Crear la visión del portafolio de proyectos.2. Identificar al Scrum Master y Stakeholder(s).3. Formar equipos Scrum.4. Desarrollar épica(s)5. Crear Backlog Priorizado del Portafolio
II. Planificación y estimación	<ol style="list-style-type: none">7. Elaborar historias de usuarios8. Estimar historias de usuario.9. Comprometer historias de usuario10. Identificar tareas11. Asignar tareas12. Crear el Sprint Backlog

III. Implementación	13. Crear entregables 14. Realizar Daily Meetings 15. Refinar el Backlog Priorizado del Portafolio 16. Demostrar y verificar el Sprint 17. Realizar Retrospectiva de Sprint.
IV. Cierre	18. Mostrar entregables. 19. Cierre del Proyecto-Retrospectiva Final.

Fuente: Elaboración propia

Estos procesos propuestos se ven reflejados en el desarrollo incremental de las actividades registradas en el software a usar:

2.2 Redmine

La herramienta Redmine es una herramienta multiplataforma con licencia GNU para la gestión y planificación de proyectos. Se trata de una solución altamente configurable que se adapta a cualquier tipo de proyecto, de manera que, la Corte Superior de Justicia pueda tener un adecuado seguimiento de tus requerimientos y control de fechas y reuniones.

Otras funciones que tiene el software Redmine:

- Es una plataforma web, por lo que puede accederse desde cualquier dispositivo independientemente del sistema operativo que se utilice.
- Está orientado a la coordinación de tareas y a la gestión de flujos de trabajo.
- Facilita y fomenta el trabajo cooperativo y la comunicación de los participantes.
- Es personalizable con el uso de complementos y temas que permiten configurar su apariencia e incrementar el número de funciones disponibles.

2.3 Entregas continuas

Puesto que se realizan entregas continuas al área usuaria, estos entregables pueden ir utilizando diferentes versiones del software que le permitan de esta manera también una constante comunicación con el área usuaria.

2.4 Modificación de requerimientos

En el software Redmine integra la funcionalidad de poder realizar un monitoreo a los requerimientos y el orden de los entregables durante el desarrollo del software.

2.5 Tiempo

El equipo de trabajo (equipo de informática) en conjunto con la aceptación del área usuaria se definió los siguientes parámetros.

Tabla 26. *Tiempos implementados para el proceso de desarrollo de software*

Tiempos	Descripción	Símbolo
Medida	Horas	H
Número de H por día	8	
Días laborables	Lunes a Viernes	
Días feriados	No laborables	
Duración de Sprint	40 horas- 5 días	40h-5D

Fuente: Elaboración propia

- **Personas y roles del proyecto**

Tabla 27. *Personas y roles del proyecto de software*

ROL	NOMBRE	CONTACTO
Scrum Master	Marco Céspedes Burga	mcespedesb@pj.gob.pe
Equipo	Diego Baldera Limo	diego.avl.04@pj.gob.pe

Product Owner (Área Usuaria)	Corte Superior de Justicia	informatica_lambayeque@pj.gob.pe
------------------------------	----------------------------	----------------------------------

3. Procedimiento

3.1 Inicío: Requerimientos y evaluación de requerimientos:

Tabla 28. *Procedimiento para el proceso de desarrollo de software*

N°	DESCRIPCIÓN DE LA ACTIVIDAD	RESPONSABLE	REGISTRO
1	Coordinar con el área de informática, la necesidad de adquirir, desarrollar o mejorar (dar mantenimiento a) un sistema o producto software; para clasificar los requerimientos del sistema o producto software.	Área solicitante	Solicitud
2	Remitir al área informática y al presidente de la Corte	Área solicitante	Memorando
3	Presidencia recibe, analiza y deriva según corresponda al área respectiva.	Presidencia de la Corte Superior de Justicia	Proveido
4	Evaluación de la viabilidad de los requerimientos: Analizar y evaluar la viabilidad técnica del	Área de Informática	Memorando

	<p>conjunto de requerimientos remitidos por el área solicitante teniendo en cuenta. Los requerimientos de infraestructura tecnológica, requerimiento de soporte y mantenimiento.</p>		
5	<p>Remitir opinión de evaluación a los requerimientos remitidos en relación con aspectos legales, operativos, de planificación y presupuesto.</p>	<p>Área de Informática</p>	<p>Informe</p>
6	<p>Si es viable continuar con el procedimiento.</p>	<p>Presidencia de la Corte Superior de Justicia</p>	<p>Informe</p>
	<p>Evaluación de opciones de adquisición Evaluación de las opciones para la adquisición del producto software requerido, de acuerdo a: a. Adquirir un producto software pre elaborado que</p>		

	<p>satisfaga los requerimientos.</p> <p>b. Desarrollar el producto software encargando su gestión al área de Informática.</p> <p>c. Desarrollar el producto software encargando su gestión a un proveedor externo.</p> <p>Los criterios mínimos a considerar en la evaluación son:</p> <p>a. Riesgos de operación y seguridad.</p> <p>b. Costos y presupuestos.</p> <p>c. Cumplimiento legal.</p> <p>d. Beneficios.</p> <p>e. Restricciones.</p>		
8	Desarrollo del producto software, encargando su gestión al área de informática.	Área de Informática	
9	Definir los alcances, responsabilidades, entregables y plazos en la obtención del producto software.	Área Usuaría	Términos o Referencia

Fuente: Elaboración propia

Planificación:

Las actividades de planificación son realizadas si la decisión de adquisición del producto software fue aprobada. El Scrum Master en conjunto con el Product Owner debe participar en las actividades que se detallan a continuación:

Tabla 29. *Planificación de actividades del desarrollo de software*

N°	DESCRIPCIÓN DE LA ACTIVIDAD	RESPONSABLE	REGISTRO
1	Elaboración del Acta de Constitución. Elaborar y suscribir conjuntamente con el Product Owner el FM01-PJ/TI: Acta de Constitución del Proyecto, es el documento que da inicio del proyecto.	Scrum Master y Product Owner	FM01-PJ/TI
2	Definición del Alcance del Proyecto. Definir, en coordinación con el Líder Usuario el alcance que tiene el proyecto y registrar en el FM02-PJ/TI: Alcance del Proyecto.	Scrum Master y Product Owner	FM02-PJ/TI
3	Elaboración de documentos del Plan de Proyecto. Elaborar los siguientes documentos que forman parte del plan de proyecto: a. FM04-PJ/TI: Plan de	Scrum Master	FM04-PJ/TI, FM05-PJ/TI

	<p>Gestión de Proyecto.</p> <p>b. FM05-PJ/TI: Estructura de Descomposición de Trabajo.</p>		
4	<p>Especificación de las Reglas de Negocio.</p> <p>Analizar y definir con el equipo de trabajo de TI en coordinación con el Product Owner las reglas de Negocio que dan soporte el producto de software solicitado, de acuerdo a los requerimientos definidos por el área usuaria</p>	Scrum Master y Product Owner	FM15-PJ/TI:
5	<p>Especificación de Requerimientos de Software.</p> <p>Analizar y elaborar con el equipo de desarrollo de TI, de acuerdo a las reglas de negocio, los requerimientos de software (funcionales y no funcionales) según los requerimientos solicitados por el área solicitante.</p>	Scrum Master y Equipo de Desarrollo	FM14-PJ/TI:
6	<p>Refinar el Backlog Priorizado del Portafolio</p> <p>El Scrum Master debe desarrollar el Product Backlog en conjunto con el Product Owner</p>	Scrum Master y Product Owner	

7	<p>Historias de Usuario</p> <p>El Scrum Master debe definir los sprints y los criterios de aceptación para el desarrollo de los entregables, para ello, debe registrar en el documento de historias de usuario FM06-PJ_TI: Historias de Usuarios</p>	Scrum Master y Equipo de Desarrollo	FM06-PJ_TI:
8	<p>Definición de la Arquitectura del Sistema.</p> <p>Analizar y definir con el equipo de proyecto de TI, el diseño de la arquitectura, que tiene el sistema, considerando los registros FM15-PJ/TI: Reglas de negocio y el FM14-PJ/TI: Especificación de requerimientos de software en el FM09-PJ/TI: Arquitectura del Sistema.</p>	Equipo de Desarrollo	FM09-PJ/TI
9	<p>Programación del producto software.</p> <p>Coordinar con el equipo de desarrollo la programación de los componentes software según el análisis y diseño del producto software.</p>	Scrum Master y Equipo de Desarrollo	

10	<p>Elaboración de documentos del producto software.</p> <p>Coordinar con el equipo de trabajo de TI, la elaboración de los documentos del producto software, tales como:</p> <p>a. FM08-PJ/TI: Manual Técnico del Sistema.</p> <p>b. FM07-PJ/TI: Manual de Usuario del Sistema.</p>	Equipo de desarrollo	FM08-PJ/TI - FM07--PJ/TI
11	<p>Pase a pruebas.</p> <p>Remitir el sistema o producto software elaborado al equipo de control de calidad, para la etapa de pruebas.</p>	Equipo de desarrollo	FM10-PJ/TI:

Fuente: Elaboración propia

Implementación:

Las actividades de implementación son realizadas luego de culminar toda la planificación y aprobación de control de calidad:

Tabla 30. *Actividades de implementación de desarrollo de software*

N°	DESCRIPCIÓN DE LA ACTIVIDAD	RESPONSABLE	REGISTRO
12	<p>Suscribir con el Líder Usuario, el FM12-PJ/TI: Acta de Conformidad de Pruebas de Aceptación del Líder Usuario.</p>	Scrum Master y Product Owner	FM12-PJ/TI:

13	Pase a producción: Proceder de acuerdo a lo definido en el FM11-PJ/TI.	Scrum Master y Equipo de Desarrollo	FM11-PJ/TI:
----	----------------------------------------------------------------------------------	-------------------------------------	-------------

Fuente: Elaboración propia

Cierre:

El Scrum Master y el Product Owner debe dejar en acta el cierre del proyecto y lo obtenido con la satisfacción del desarrollo:

Tabla 31. *Actividades de cierre del desarrollo de software*

N°	DESCRIPCIÓN DE LA ACTIVIDAD	RESPONSABLE	REGISTRO
14	Cierre de proyecto. Suscribir conjuntamente con el Product Owner el FM13-PJ/TI: Acta de Cierre del Proyecto. Nota: La documentación generada durante la ejecución del proyecto, debe encontrarse revisada, aprobada y firmada antes del cierre del proyecto.	Scrum Master y Product Owner	FM13-PJ/TI:

Fuente: Elaboración propia

Anexo 7.2. Tiempo promedio de entrega de desarrollo de software.

Redmine es una herramienta para la gestión de proyectos, que con sus diversas funcionalidades permite a los usuarios de diferentes proyectos realizar el seguimiento y organización de los mismos. Además, es posible optimizar su funcionamiento agregando funcionalidades. Otras herramientas que incluye son calendario de actividades, diagramas de Gantt para la representación visual de la línea del tiempo de los proyectos, wiki, foro, visor del repositorio de control de

versiones, RSS, control de flujo de trabajo basado en roles, integración con correo electrónico, entre otras opciones.

Figura 6. Gestión de peticiones

Tipos de peticiones

Tipo	Estado Predeterminado	Descripción	
Tareas	Pendiente	Asignación de Tareas a los participantes del equipo	Copiar Borrar
Desarrollo	Pendiente		Copiar Borrar
Reunión	Pendiente	Reuniones de Planificación.	Copiar Borrar
Sprint	Pendiente	Sprint del proyecto	Copiar Borrar

Nota: El gráfico se visualiza los tipos de peticiones considerando las tareas que se le asigna a cada miembro del equipo, las tareas para desarrolladores, el seguimiento y control de reuniones y la creación de los sprints. *Elaboración propia*

Figura 7. Estado de Peticiones

Estados de las peticiones

Estado	Petición resuelta	
Nueva		Borrar
En curso		Borrar
Resuelta		Borrar
Comentarios		Borrar
Cerrada	✓	Borrar
Rechazada	✓	Borrar

Nota: El gráfico se visualiza los Estados de las peticiones. *Elaboración propia*

Figura 8. Estado de peticiones durante el Desarrollo del Software

Inicio MI página Proyectos Administración Ayuda

Sistema de Denuncias

Conectado como administrador | Mi cuenta | Terminar sesión

Búsqueda: Sistema de Denuncias

Vistazo Actividad Peticiones Tiempo dedicado Gantt Calendario Noticias Documentos Wiki Archivos Configuración

Peticiones Nueva petición ...

Filtros: Estado todos Añadir el filtro

Opciones: Aceptar Anular Guardar

Tempo estimado: 71.00 Tempo dedicado: 0.00

#	Tipo	Estado	Prioridad	Asunto	Asignado a	Fecha de inicio	Fecha fin	% Realizado
Pendiente Tempo estimado: 19.00 Tempo dedicado: 0.00								
63	Tareas	Pendiente	Normal	[HUI] Iniciar Sesión - Login	Eduardo Gonzales	2021-12-18	2021-12-20	0%
62	Tareas	Pendiente	Normal	Sprint 01	Eduardo Gonzales	2021-12-15	2021-12-18	100%
57	Tareas	Pendiente	Normal	Elaborar documento de acta de constitución	Sara Sinfuentes	2021-12-02	2021-12-03	0%
En progreso Tempo estimado: 3.00 Tempo dedicado: 0.00								
61	Tareas	En progreso	Alta	Elaborar documento de historias de usuarios	Diego Baldera Lino	2021-12-13	2021-12-15	50%
Finalizado Tempo estimado: 49.00 Tempo dedicado: 0.00								
60	Reunión	Finalizado	Normal	Reunión de Planificación	Franco Damiani	2021-12-03	2021-12-03	100%
59	Tareas	Finalizado	Normal	Elaborar documento de plan de gestión del proyecto	Sara Sinfuentes	2021-12-07	2021-12-07	100%
58	Tareas	Finalizado	Normal	Elaborar documento de alcance del proyecto	Sara Sinfuentes	2021-12-03	2021-12-06	100%
56	Tareas	Finalizado	Normal	Kick Off del Proyecto	Franco Damiani	2021-12-01	2021-12-01	100%

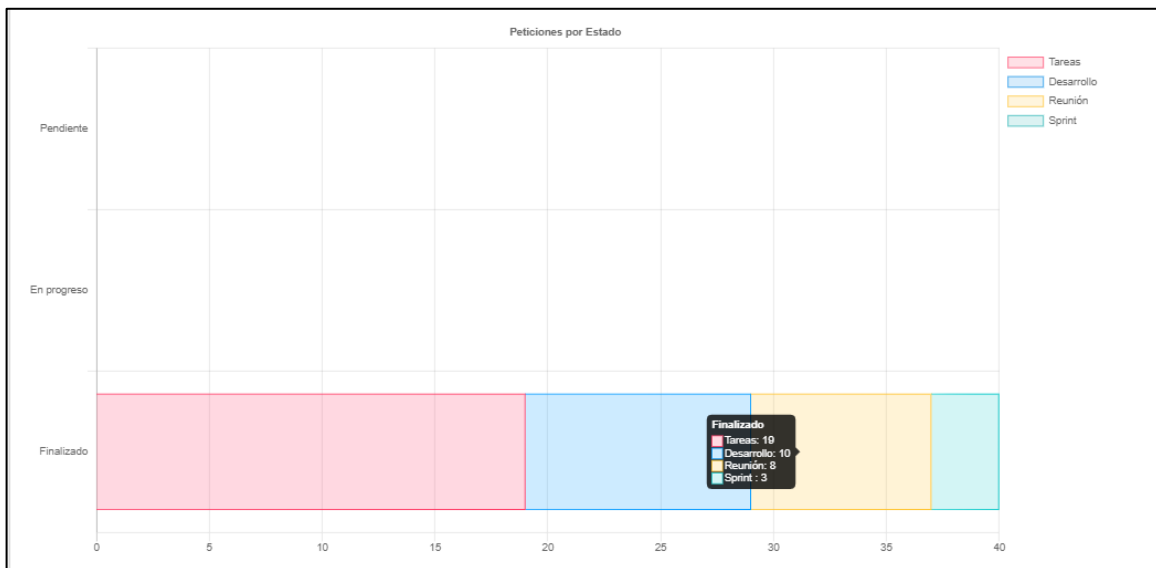
(1-8/8)

Exportar a: Atom | CSV | PDF

Consultas personalizadas:
 Peticiones actualizadas
 Peticiones monitorizadas
 Peticiones que me están asignadas
 Peticiones registradas por mí

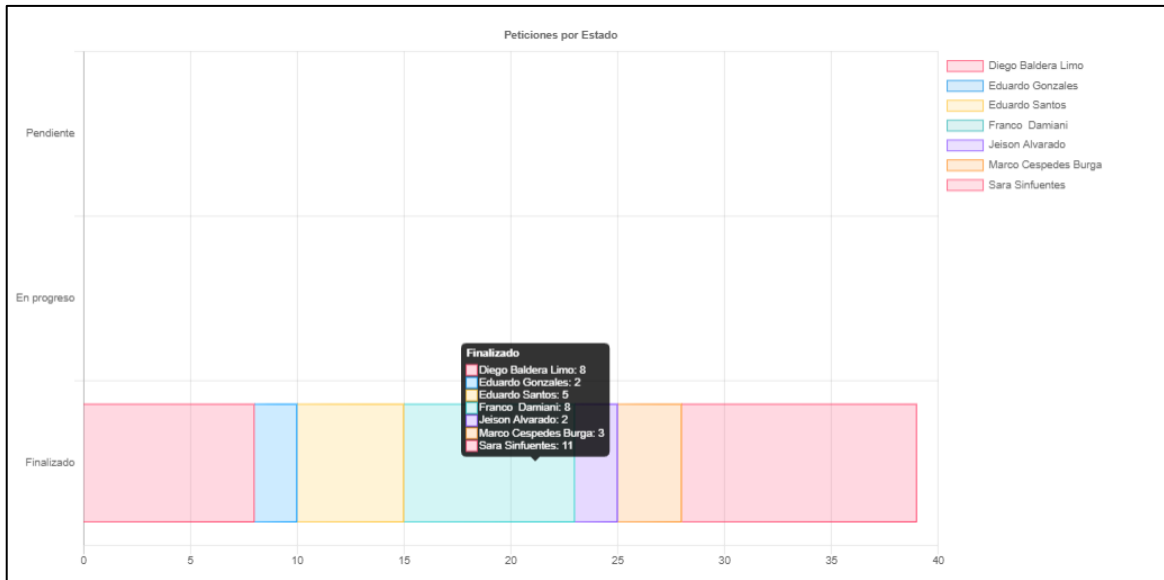
Nota: El sistema le permite al usuario ver los trabajos que se encuentran en diferentes estados pendiente, en progreso y finalizado. *Elaboración propia*

Figura 9. Resumen del desarrollo de Software – Grafico estado de peticiones



Nota: El sistema le permite al usuario ver la cantidad de peticiones desarrolladas durante el desarrollo de software diferenciándola por tipo. *Elaboración propia*

Figura 10. Resumen del desarrollo de Software – Requerimientos por usuarios

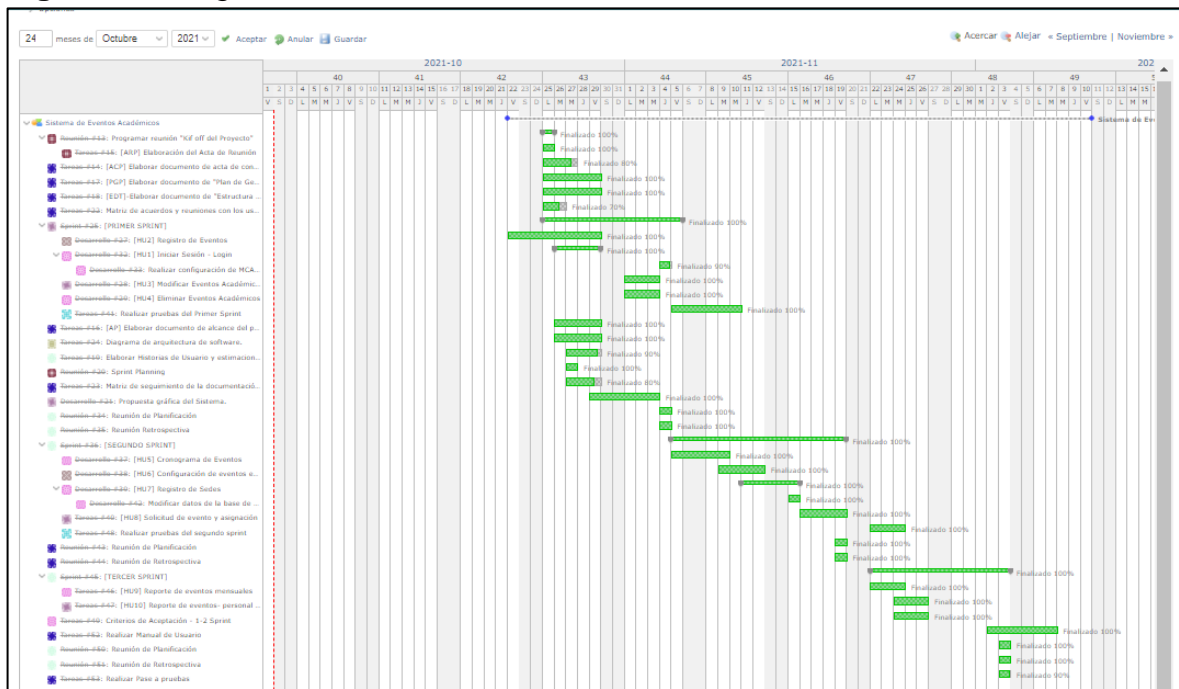


Nota: El sistema le permite al usuario ver el resumen del desarrollo de software que se ejecutó.

Elaboración propia

Anexo 7.3. Nivel de comunicación con el área usuaria durante el desarrollo del software

Figura 10. Diagrama de Gantt



Nota: La gestión de tareas es uno de los puntos fuertes de Redmine. Los distintos procesos y tareas pueden ser creados, asignados a usuarios, modificados, monitorizados y evaluados en tiempo real, permitiendo un control total de todo lo que ocurre durante su ciclo de vida y demostrados en esta línea de tiempo. *Elaboración propia.*

Anexo 7.4. Nivel de satisfacción con el área usuaria

Figura 11. *Evaluación de nivel de satisfacción con el área usuaria*

Evaluación de avance de Sprint

En este formulario el área usuaria podrá realizar una evaluación del avance del Proyecto de Sistema de Gestión de Riesgos.

Hola, Jessica. Cuando envíe este formulario, el propietario verá su nombre y dirección de correo electrónico.

* Obligatorio

1. ¿El sistema está cumpliendo con el objetivo planificado? *

1 2 3 4 5

2. ¿El equipo de desarrollo está cumpliendo con la funcionalidad planificada? *

1 2 3 4 5

3. ¿Cuál es tu opinión sobre la organización de la información en las pantallas? *

Muy confuso

Confuso

Claro

Entendible

4. ¿Tiene alguna sugerencia para la próxima fase de desarrollo (incluida la presentación)? *

Escriba su respuesta

Enviar

Nota: La figura muestra la evaluación realizada para medir el nivel de satisfacción con el área usuaria. *Elaboración propia.*