

INNOVACIÓN EN EL DESARROLLO DE SOFTWARE PARA OPTIMIZAR EL SEGUIMIENTO DE ACUERDOS

INNOVATION IN SOFTWARE DEVELOPMENT TO OPTIMIZE AGREEMENT TRACKING

Xochitototl Cote Itzel¹, Hernández Mora José Juan², Medina Barrera María Guadalupe³, Cuatecontzi Cuahutle Elizabeth⁴, Ramos Ramos Juan⁵

<https://doi.org/10.61117/ipsumtec.v8i1.351>

¹Ingeniera en Gestión empresarial. Tecnológico Nacional de México/IT de Apizaco. División de Estudios de Posgrado e Investigación. Maestría en Sistemas Computacionales. m23370012@apizaco.tecnm.mx. Tel: 241 4172010 Ext. 146. Av C.P. 90300. <https://orcid.org/0009-0008-1409-549X>

²Doctor en Excelencia Docente. Tecnológico Nacional de México/ IT de Apizaco. División de Estudios de Posgrado e Investigación. Maestría en Sistemas Computacionales. juan.hm@apizaco.tecnm.mx. Tel: 241 4172010 Ext. 146. Av C.P. 90300. <https://orcid.org/0000-0003-2878-7290>

³Doctora en Planeación estratégica y Dirección Tecnológica. Tecnológico Nacional de México/ IT de Apizaco. División de Estudios de Posgrado e Investigación. Maestría en Sistemas Computacionales. guadalupe.mb@apizaco.tecnm.mx. Tel: 241 4172010 Ext. 146. Av C.P. 90300. <https://orcid.org/0000-0003-3074-0029>

⁴Maestra en Dirección de Ingeniería de Software. Tecnológico Nacional de México/ IT de Apizaco. División de Estudios de Posgrado e Investigación. Maestría en Sistemas Computacionales. elizabeth.cc@apizaco.tecnm.mx. Tel: 241 4172010 Ext. 146. Av C.P. 90300. <https://orcid.org/0009-0008-0531-0354>

⁵Doctor en Sistemas Computacionales. Tecnológico Nacional de México/ IT de Apizaco. División de Estudios de Posgrado e Investigación. Maestría en Sistemas Computacionales. juan.rr@apizaco.tecnm.mx. Tel: 241 4172010 Ext. 146. Av C.P. 90300. <https://orcid.org/0009-0004-7440-7257>

Resumen -- En el ámbito del desarrollo de software, el aseguramiento de la calidad y la gestión eficaz de los hallazgos son aspectos cruciales para el éxito de cualquier proyecto. Además de documentar los hallazgos identificados, es fundamental llevar a cabo un seguimiento detallado para asegurar que se hayan implementado las correcciones necesarias [1]. Por ello, en este trabajo, se presenta un formato innovador para el seguimiento de acuerdos en el desarrollo de software, el cual está diseñado para facilitar la revisión, validación y verificación de las actividades.

Es por ello, que se diseña un formato para la gestión de acuerdos en proyectos de desarrollo de software, cubriendo distintas etapas del ciclo de vida del proceso. Posteriormente, su implementación fue analizada mediante gráficas de anillos, para el seguimiento de las actividades. Los resultados de los casos de estudio demuestran que esta implementación mejora considerablemente la organización del proceso de desarrollo.

En resumen, en este trabajo se manifiesta que el formato propuesto es una herramienta efectiva para evitar que las actividades del ciclo de un proyecto de software queden inconclusas, mejorando así la calidad y el éxito de los proyectos de desarrollo de software.

Palabras Clave: Comunicación, Desarrollo de software, Gestión del software, Validación, Verificación.

Abstract -- In the field of software development, quality assurance and the effective management of findings are crucial aspects for the success of any project. In addition to documenting the identified findings, it is essential to conduct detailed follow-up to ensure that necessary corrections have been implemented [1]. Therefore, this paper presents an innovative format for tracking agreements in software development, designed to facilitate the review, validation, and verification of activities.

Thus, a format is designed for managing agreements in software development projects, covering different stages of the process life cycle. Its implementation was then analyzed using ring graphs to track activities. The results from the case studies demonstrate that this implementation significantly improves the organization of the development process.

In summary, this paper shows that the proposed format is an effective tool for ensuring that activities in a software project cycle do not remain incomplete, thereby improving both the quality and success of software development projects.

Key words – Communication, Software development, Software management, Validation, Verification.

INTRODUCCIÓN

Los estudios realizados por Quitian Monroy JS, et al [2] muestran que, en el desarrollo de software con metodologías ágiles, la comunicación es la práctica más importante. Sin embargo, los equipos de desarrollo autónomos operan al 50% de su capacidad. En contraste, en metodologías tradicionales la comunicación se lleva a cabo en un 76% de los casos y los equipos de desarrollo autónomos trabajan al 32% de su capacidad, esto indica que el flujo de trabajo no es constante y es necesario proporcionar una comprensión del rol y las actividades específicas a desarrollar por cada participante en el proceso, lo que permite la optimización y automatización del proceso [3], como se puede observar en la tabla 1.

Tabla 1. Cuadro de ponderación

Ágiles	Tradicionales	
Comunicación	100% Integración de tecnología	80%
Grupo multifuncional	95% Integración del cliente	80%
Versatilidad	85% Comunicación	76%
Integración de tecnología	80% Innovación	56%
Integración del cliente	75% Grupo multifuncional	56%
Simplicidad	60% Mejora continua	52%
Innovación	55% Versatilidad	48%
Grupo de desarrollo autónomo	55% Simplicidad	44%
Mejora continua	55% Grupo de desarrollo autónomo	32%

Fuente. Adaptado de Quitian Monroy JS, Martínez Padua JV, Castiblanco Jiménez IA [2].

El formato que se propone es para innovar y registrar la forma en cómo se comunica el equipo de desarrollo de software, así mismo, documenta los acuerdos que se toman en las reuniones relacionadas a un proyecto de software específico, con el objetivo de que el equipo de desarrollo refuerce su aprendizaje en relación con los proyectos, tareas y actividades asignadas como parte de su trabajo, se promueve compartir los errores cometidos durante el desarrollo de los procesos para que todos los involucrados puedan aprender de ellos [1].

Además, un flujo de trabajo monitoreado permite a los empleados tomar decisiones oportunas, mejorar la comunicación del equipo e impulsar el trabajo ágil en los procesos de desarrollo de software [3]. Todo esto se lleva a cabo bajo una administración rigurosa de la documentación de actividades, informes de monitoreo, cambios, riesgos operativos y de seguridad, así como los impactos de estos riesgos, lo que permite un monitoreo adecuado y garantiza la disponibilidad continua de los servicios y activos de información [4], [5].

Problemática

Ante la necesidad de documentar y monitorear las actividades durante el proceso de desarrollo de software, se han identificado obstáculos en la coordinación del equipo [6]. La metodología ágil, como Kanban, ofrece soluciones en la manufactura de software, pero depende fundamentalmente de un equipo bien coordinado, comunicado, responsable y cohesionado [7]. Por lo tanto,

es esencial realizar un seguimiento continuo de su implementación.

Así mismo, la metodología Scrum gestiona rigurosamente los procesos para generar valor al cliente, por lo tanto, es importante identificar las tareas, para adelantarlas si es posible, pero esta práctica debilita los indicadores de calidad de la tarea realizada, debido al dominio de la limitación del trabajo en curso, además, se pueden producir cuellos de botella durante las etapas de desarrollo. Por ejemplo, Lean [8] propone realizar sólo entrega de recomendaciones, las cuales son instrucciones no específicas. Esto da como resultado la subjetividad a la interpretación de cada equipo de trabajo.

Por lo tanto, es esencial evaluar el seguimiento de los proyectos, adaptar la planificación, organizar el trabajo de manera eficiente y mantener una comunicación efectiva dentro del equipo para lograr los mejores resultados y maximizar el rendimiento [9].

Además, un estudio realizado por Torres en 2024 [10] demuestra que, entre las buenas prácticas más destacadas en las empresas de desarrollo de software, solo el 2,56% verifica los avances con los *stakeholders*.

Por su parte, Merchán-Narváez N J, et al. [11] revela que las metodologías ágiles enfrentan problemas durante la implementación, como la falta de comunicación efectiva, lo que impacta negativamente en su eficacia. Los procesos dependen en gran medida del gestor del proyecto y sin un monitoreo adecuado, las actividades pueden fracasar. Por otro lado, la autorregulación puede dar lugar a equipos indisciplinados si no están comprometidos con sus tareas y estas no se verifican de manera constante. Además, la participación del cliente en las revisiones del desarrollo puede convertirse en un desafío si el equipo no está plenamente comprometido.

Asimismo, la falta de un control de versiones adecuado, una gestión de pruebas técnicas insuficiente y un deficiente aseguramiento de la calidad en los proyectos de desarrollo de software evidencian una necesidad urgente de documentación, verificación y monitoreo continuo de las actividades realizadas durante el proceso [4].

Alcances y limitaciones del formato

Por otro lado, la asignación e identificación de los responsables de los procesos contribuye a una mejor aplicación del flujo general de trabajo [12]. Esta asignación facilita la automatización y optimización de las actividades realizadas y verificadas, garantizando que los colaboradores cumplan con sus tareas de manera oportuna.

En otras palabras, un flujo de trabajo constante y automatizado debería, en teoría, no obstaculizar los procesos de desarrollo de software. Sin embargo, en ocasiones puede haber problemas si los responsables no cumplen con los acuerdos establecidos, lo que puede afectar la calidad del producto. Además, la falta de procesos definidos en las empresas puede dificultar la verificación y el monitoreo general del producto [13] y [14].

Beneficios del formato

- Mejorar la comunicación del equipo de desarrollo.
- Claridad en los objetivos de cada actividad.
- Comprometer a los colaboradores a realizar sus actividades.
- Integrar evidencia de la revisión y verificación de las actividades.
- Evitar malentendidos entre las partes interesadas.

Desventajas de la implementación del formato

Durante la implementación de este formato, se encontraron inconvenientes relacionados a:

-La curva de aprendizaje y la adaptación al nuevo formato afectaron la productividad a corto plazo como sucedió en el caso uno que se muestra más adelante.

-La resistencia al cambio, en donde los colaboradores no estaban cómodos con la revisión constante provocando que el formato fuera poco eficaz, como sucedió en el caso dos que se muestra más adelante.

-Además, se creó un sentimiento de sobre carga administrativa, que fue el resultado de la documentación y seguimiento donde antes no se realizaba, como sucedió en el caso tres que se muestra más adelante.

Generalidades

Considerando los aportes de Helen Tupper y Sarah Ellis [3], Gramuntell Bayarri V [5], IBM [9], Torres Calle [10] y Fernando Villafranca A [15], se ha diseñado el formato con los siguientes principios:

-Funcionalidad: gestión de tareas, seguimiento de errores, informes y análisis, lista de verificación, seguimiento de comentarios, colaboración en tiempo real.

-Determinar un proceso a correlacionar: es la forma en cómo se va a relacionar el formato en el flujo de trabajo del proyecto en cuestión. Los identificadores deben de ser únicos, para evitar confusiones.

-Información que identifica al proyecto, al equipo y al evento en cuestión: con el fin de que todas las partes interesadas estén en la misma frecuencia y no haya espacio para las ambigüedades.

-Pasos del flujo de trabajo: se deben de identificar los pasos clave, entradas, resultados, métricas, personal involucrado, sus áreas de trabajo, así como sus responsabilidades dentro del proyecto a desarrollar y proceso de verificación.

-*Feedback* y verificación: se analiza la validación de los comentarios de las partes interesadas, con la finalidad de identificar errores y cuellos de botella recurrentes durante la fase de producción con la finalidad de mejorar los procesos.

-Ajustes de retroalimentación: con la finalidad de regular y automatizar el funcionamiento del flujo de trabajo dentro del equipo de producción de software, de esta manera lograr el progreso individual y grupal de las actividades a realizar.

DESARROLLO

El presente trabajo se realizó en una empresa de desarrollo de software internacional, que por motivos de confidencialidad no es posible mencionar el nombre, así como en el Centro de Cómputo de la Maestría en Sistemas Computacionales del Tecnológico Nacional de México/Campus Apizaco.

Formato

Tomando en cuenta las particularidades anteriores y las aportaciones realizadas por Helmut Sy Corvo. [17] y Peraza-Delgado M [18], quienes proponen la utilización de formatos para el seguimiento de acuerdos y pruebas respectivamente; en la propuesta de formato realizada en este proyecto, se agrega una sección de validación de la información, la cual permitirá el correcto seguimiento de las actividades con la finalidad de que éstas no queden inconclusas.

El formato propuesto se muestra en la Tabla 2 y consta de cuatro secciones: la primera proporciona datos de identificación, la segunda detalla los objetivos, la tercera enumera los acuerdos y la cuarta se centra en la validación de la información. Este último apartado es el más crucial para garantizar el éxito de las actividades relacionadas con los acuerdos establecidos.

Tabla 2. Formato de seguimiento de acuerdos en el desarrollo de software.

Seguimiento de Acuerdos en el Desarrollo de Software	
Proyecto:	
Equipo de desarrollo:	
Fecha:	
Hora de inicio de la reunión:	
Hora de finalización de la reunión:	
Objetivos:	
1.-	
2.-	
3.-	
Acuerdos:	
Acuerdo 1:	
Acuerdo establecido:	
Nombre del responsable:	
Constancia de notificación al responsable:	
Fecha de entrega del acuerdo:	
Acuerdo 2:	
Acuerdo establecido:	
Nombre del responsable:	
Constancia de notificación al responsable:	
Fecha de entrega del acuerdo:	
Acuerdo 3:	
Acuerdo establecido:	
Nombre del responsable:	
Constancia de notificación al responsable:	
Fecha de entrega del acuerdo:	
Validación de información.	
Relaizado por:	
Firma electrónica:	
Persona responsable que valida:	
Firma electrónica:	
Responsable que verifica:	
Firma electrónica:	

Fuente. Creación propia.

Aplicación del formato en entorno de desarrollo de software.

Para probar la eficacia y demostrar los beneficios del formato propuesto se implementa en el desarrollo de una aplicación para la gestión del talento humano. Es importante destacar que este formato es compatible tanto con metodologías ágiles como tradicionales de desarrollo de software, como se ilustra en las figuras 1 y 2.

Figura 1. Parte 1 de la aplicación del formato.

Proyecto: Convenio Tecnológico Nacional de México Campus Apizaco - Algar Tech México.
Equipo que desarrolla: Maestría Generación 18
Fecha: 11/10/2023
Hora de inicio: 14:00 hr.
Hora de finalización: 14:30 hr.

Objetivos de la reunión:

- 1.- Seguimiento de proyectos de convenio ITA - Algar Tech
- 2.- Revisión de actividades de los estudiantes de Maestría en la empresa AlgarTech.
- 3.- Revisión de actividades de proyecto Tiquetera

Asuntos:

Asunto 1: Requerimientos adicionales
Solicita buscar, ya que al buscar una persona en la base de datos de los asociados de Algar que solicitan permisos, se hace de forma manual.

N°	Cédula	Nombre	Correo	País	Contrato	Activo	Cédula Líder
637				México		☒	
635				México		☐	0
679				Colombia		☒	
794				México		☒	
809				Colombia		☒	
925				Colombia		☒	
1175				Colombia		☒	

Actualizar Anterior 1 25 26 32 180 Siguiente

Proceso del registro de colaboradores en formato .xlsx

Registro de empleados

Consultar empleados Elegir archivo Ninguno archivo seleccionado

Fuente. Creación propia.

La figura 1, muestra el ejemplo de llenado de la información, el nombre del proyecto, el equipo asignado al proyecto, la fecha y la hora de inicio y termino de la reunión. Posteriormente, se muestran los objetivos de la reunión y el primer acuerdo tomado, junto con la evidencia que corresponde al primer asunto.

Figura 2. Parte 2 de la aplicación del formato.

Nombre del responsable: Todo el equipo.
Evidencia de notificación al responsable: Todo el equipo estuvo presente en la reunión.
Fecha de entrega del asunto: Diciembre de 2023.

Asunto 2: Funcionalidad actual
Proporcionar video y manual de usuario de la tiquetera. Instructivo con video.
Nombre del responsable: María Camila
Evidencia de notificación al responsable: Camila estuvo presente en la reunión.
Fecha de entrega del asunto: A más tardar 13 de octubre de 2023.

Asunto 3: Documentación del proyecto
Itzel genera y se encarga que todo el equipo tenga a su disposición la información del proyecto. Crear equipo en teams. Plan de trabajo, RoutMap, Presentaciones, Historias de usuario, etc. Llenado de Planner.
Nombre del responsable: Itzel Xochitototl Cote
Evidencia de notificación al responsable: Itzel estuvo presente en la reunión.
Fecha de entrega del asunto: 12 de octubre de 2023.

Asunto 4: Decisión de tareas
Reunión de equipo para establecer entornos de desarrollo y mejorar la comunicación.
Nombre del responsable: Todo el equipo.
Evidencia de notificación al responsable: El equipo estuvo presente en la reunión.
Fecha de entrega del asunto: 12 de octubre de 2023.

Validación de información.

Hecho por: Itzel Xochitototl
Firma electrónica: [Firma]
Persona responsable que valida: Carolina
Firma electrónica: [Firma]
Responsable que verifica: Cliente
Firma electrónica: [Firma]

Fuente. Creación propia.

La figura 2, muestra el complemento de la sección de asuntos, la especificación del responsable de llevar a cabo las actividades, evidencia de que se le notificó y la fecha en que el acuerdo debe de ser entregado, al final se encuentra el apartado de validación de la información y

es lo más importante para garantizar el éxito de las actividades y el logro de los objetivos estipulados en el presente documento. A partir de este seguimiento, se pueden realizar las gráficas de anillos, para identificar el progreso de las actividades y verificar que se culminen.

DISCUSIÓN Y ANÁLISIS DE RESULTADOS

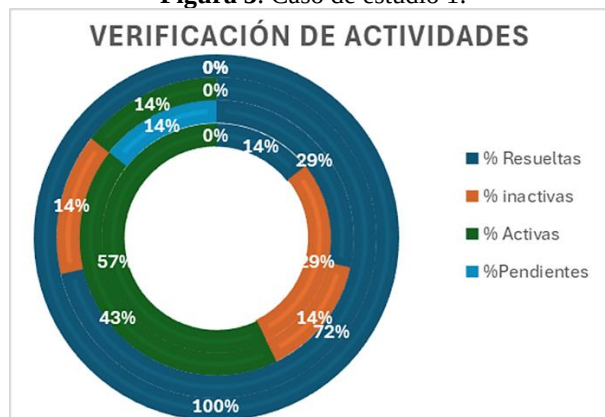
Resultados de la aplicación del formato en entorno de desarrollo de software mediante casos de estudio.

Tomando en cuenta el formato anterior, se muestra la gráfica del progreso y revisión de las actividades estipuladas en el formato; en tres casos de estudio.

Caso de estudio Uno

La figura 3, muestra el seguimiento de los acuerdos mostrados en el ejemplo de aplicación (figuras 1 y 2).

Figura 3. Caso de estudio 1.



Fuente. Creación propia.

La figura 3, representa el progreso de las actividades supervisadas, este seguimiento permitió que se mejorara la aplicación en desarrollo y los acuerdos tomados se concluyeran después de cuatro revisiones. Donde, la primera iteración de la gráfica muestra que el 14% de las actividades están resueltas, el 29% de las actividades están inactivas, es decir, no se han podido iniciar, el 57% se están realizando, por lo que tienen la etiqueta de activas y no se encuentran actividades pendientes. Durante la segunda revisión de avances, el 29% de las actividades ya están resueltas, aún hay actividades que no se ha iniciado y representan el 14%; el 43% de las actividades se están llevando a cabo y se encontraron inconsistencias en el 14% de las actividades, por lo que se identificaron como pendientes. Mientras, en la tercera supervisión de las actividades, el 72% ya están terminadas y aprobadas, el 14% están inactivas, el 14% están activas y no hay actividades pendientes. Con este seguimiento, todo el equipo logró terminar sus actividades, incluyendo la inconsistencia, que, de no tener seguimiento de verificación y validación, hubiera quedado inconclusa. Cabe destacar, que, en este proyecto, la inconsistencia tuvo su origen en la curva de aprendizaje que se implementó en todo el equipo de desarrollo, sin embargo,

el correcto seguimiento y la comunicación subsanaron el detalle.

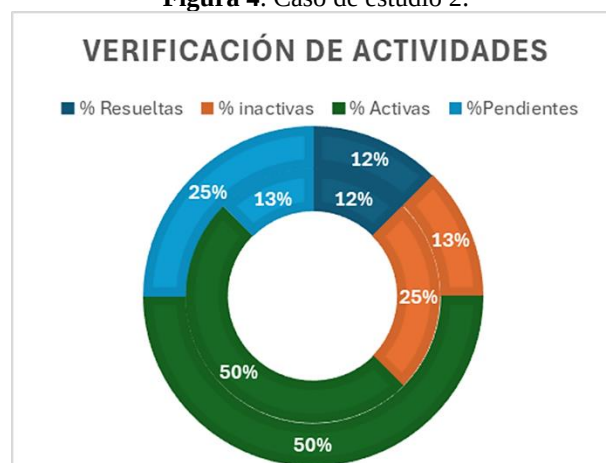
Esto indica que la validación y verificación son importantes al momento de desarrollar software de calidad y con las especificaciones que el cliente solicita.

Caso de estudio Dos

En este caso, se observó una notable resistencia al cambio por parte del equipo. Los miembros se sentían incómodos con la validación y verificación constante de las actividades, y mostraron descontento con la implementación de los nuevos procesos. Esta resistencia hizo que el formato propuesto para el seguimiento de acuerdos resultara menos efectivo y condujera a una ejecución deficiente, afectando negativamente su eficacia.

Además, los cambios en el alcance del proyecto no se gestionaron adecuadamente, lo que llevó a actividades inconclusas, como se muestra en la figura 4. Esta figura, también, ilustra cómo una revisión insuficiente de los avances puede llevar al incumplimiento de los acuerdos establecidos.

Figura 4. Caso de estudio 2.



Fuente. Creación propia.

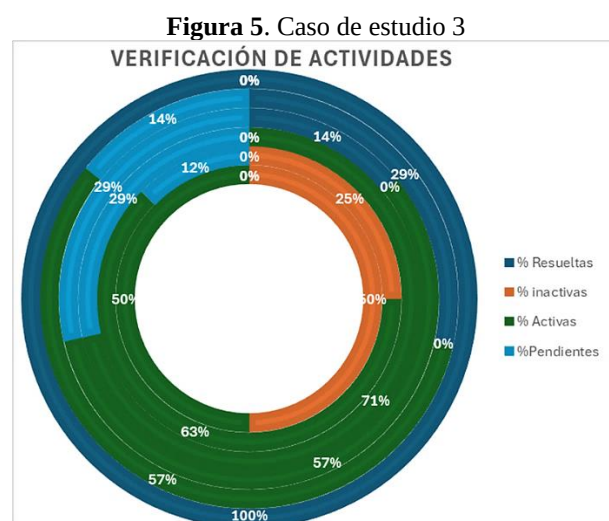
Tomando en consideración los argumentos anteriores, en la figura 4, se observa que, durante la primera revisión, el 12% de las actividades estaban resueltas, el 25% estaban inactivas, el 50% activas y el 13% de las actividades inactivas durante la primera iteración del formato.

Posteriormente, al realizar la segunda iteración, no se avanzó con las actividades resueltas y continuaron solo el 12%, lo que nos proporcionó un indicador de que algo no funcionó correctamente, el 13% de las actividades continuaron inactivas, con lo que se muestra que las activas debieron haber aumentado, pero no lo hicieron; finalmente, las actividades pendientes con inconsistencias aumentaron a 25%, dando como resultado que las actividades del caso dos quedaran inconclusas.

Con todo ello, se demuestra que el formato de revisión también sirve como una herramienta esencial para validar la calidad del software, ya que proporciona un registro de los avances en relación con los acuerdos establecidos. Sin embargo, la resistencia al nuevo formato impidió su uso efectivo, con ello se hace énfasis en la necesidad de una mayor aceptación y adaptación a las nuevas prácticas de seguimiento.

Caso de estudio tres

Para demostrar la compatibilidad del formato con diferentes metodologías, la Gráfica 3 muestra el seguimiento de las actividades durante el diseño de una aplicación web para seguimiento escolar. En este caso, se consideraron tanto los acuerdos iniciales como las tareas asignadas a cada miembro del equipo.



Fuente. Creación propia.

La figura 5 muestra que, al inicio de las actividades, había 50% activas y 50% inactivas, en la segunda iteración 25% inactivas, 63% activas, 12% pendientes y no se reportan actividades resueltas; para la tercera iteración, ya no hubo actividades inactivas, 71% activas y las pendientes aumentaron al 29%, relacionadas a inconsistencias en las actividades realizadas, así mismo, no se reportaron actividades resueltas; para la cuarta iteración, el 14% de las actividades quedaron concluidas, 57% activas, las actividades pendientes con inconsistencias no disminuyeron, pero tampoco aumentaron y seguían sin reportarse actividades resueltas; para la quinta iteración, el 29% de las actividades ya estaban resueltas, ya no hubo inactivas, el 57% de las actividades estaban activas y las actividades pendientes disminuyeron al 14%; se puede observar que la revisión constante dio como resultado para la sexta iteración, el 100% de las actividades concluidas y aceptadas, a pesar de que, en el equipo se generó un sentimiento de exceso de trabajo, originado en las revisiones y validaciones constantes.

Con ello se manifiesta que, para lograr los objetivos establecidos, se deben de llevar a cabo las revisiones y

validaciones, demostrando que la aplicación del formato propuesto es un trabajo constante y asegura la calidad del producto entregado.

CONCLUSIONES

La implementación del formato propuesto permite identificar claramente los roles y responsabilidades, lo que previene la duplicación de actividades. Esto facilita la planificación, la comunicación y el aseguramiento de la calidad, ya que el formato ofrece una visión clara de las tareas y un camino definido para su seguimiento. Además, la falta de comunicación y las prácticas inconsistentes pueden llevar a desacuerdos. Por lo tanto, una supervisión constante es crucial para reducir los efectos de la improvisación excesiva y la falta de control [19]. Se ha desarrollado un proceso para gestionar la mantenibilidad desde las primeras etapas del ciclo de vida del software, especificando las entradas y salidas para cada actividad asignada a roles determinados [20].

Este proceso incluye roles para verificación y validación de la mantenibilidad. Y es ahí donde el formato propuesto toma el papel fundamental como herramienta de apoyo y seguimiento.

Por otro lado, se presentaron inconsistencias durante la implementación; dichas inconsistencias se abordaron con una planificación cuidadosa y una comunicación clara, con ello se logró minimizar el impacto negativo y asegurar una transición exitosa hacia la implementación del formato propuesto.

En resumen, el formato propuesto no solo organiza de manera efectiva las actividades y roles, sino que también fortalece la planificación y el control del proyecto. La implementación de un enfoque detallado para la verificación y validación de la mantenibilidad en consecuencia se asegura una gestión integral y eficaz del ciclo de vida del software.

Como investigaciones futuras, se propone la utilización del formato en la gestión de desarrollo de software bajo una metodología híbrida de desarrollo de software denominada "ÁgilSegura" con base en DevSecOps y SCRUM.

Las líneas de trabajo futuras podrían centrarse en:

- La optimización de la implementación de metodologías ágiles con un formato estructurado y sobre cómo integrar dicho formato dentro de los marcos ágiles más populares, sin imponer una estructura rígida que pueda afectar la adaptabilidad del equipo.

- Mejora de la verificación de la mantenibilidad en el ciclo de vida del software, para identificar cómo estas prácticas pueden evolucionar a lo largo del ciclo de vida del proyecto. Se podrían explorar nuevas métricas de calidad específicas para la mantenibilidad, así como herramientas

de automatización que integren la verificación continua durante todo el proceso de desarrollo, asegurando que el software sea fácil de mantener, modificar y expandir con el tiempo.

-El impacto de la supervisión constante y el control en el éxito del proyecto, pues puede parecer contraproducente en un entorno ágil, además se podrían investigar métodos y herramientas específicas para integrar la supervisión en equipos ágiles sin generar sobrecarga administrativa, garantizando que el proyecto se mantenga enfocado, dentro del presupuesto y los plazos establecidos.

AGRADECIMIENTOS

Los autores agradecen al Tecnológico Nacional de México Campus Apizaco, División de estudios de posgrado e investigación por el préstamo del laboratorio y del equipo para realizar este Proyecto, a la Empresa de Desarrollo de Software Internacional, por abrir sus puertas y permitir el desarrollo de la investigación.

BIBLIOGRAFÍA

[1] Dirección General de Cómputo y de Tecnologías de Información y Comunicación. (2022) “Prácticas para el desarrollo de software” [Online] Dirección general de cómputo y de tecnologías de información y comunicación. Universidad Autónoma de México UNAM [Consultado el 05 de agosto de 2024] Disponible en: <https://www.red-tic.unam.mx/desarrollo-de-software>

[2] Quitian Monroy JS, Martínez Padua JV, Castiblanco Jiménez IA. (2022) “Caracterización y comparación de metodologías ágiles y tradicionales de desarrollo de producto”. [Online] Cien.Ing. Neogranadina [Internet]. ;32(2):9-26. [Consultado 19 de agosto de 2024] Disponible en: <https://doi.org/10.18359/rcin.5168>

[3] Helen Tupper y Sarah Ellis, (2023) “How to help your team learn in the Flow of work”, [Online] Harvard Business Review February. [Consultado el 05 de agosto de 2024] Disponible en: <https://hbr.org/2023/02/how-to-help-your-team-learn-in-the-flow-of-work>

[4] Pinzón López L M, García Rodríguez J C, (2024) “Reestructuración del proceso de control de cambios de software para una institución de educación superior” [Online] Trabajo de grado. Universidad de Bosque [Consultado el 05 de agosto de 2024] Disponible en: <https://hdl.handle.net/20.500.12495/12693>

[5] Gramuntell Bayarri, Vicente, (2023) “Análisis y comparativa de herramientas de gestión de proyectos” [Online] Trabajo de grado. Universidad Politécnica de Valencia, [Consultado el 05 de agosto de 2024] Disponible en: <http://hdl.handle.net/10251/197137>

[6] ALMAGUER TORRES R M, PEREZ CAMPANA M y AGUILERA GARCIA L O, (2020) “Procedimiento para la gestión integrada y por procesos de proyectos de desarrollo local.” Rev retos [online]., vol.14, n.1 [Consultado 2024-08-22], pp. 89-115 . Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2306-91552020000100089&lng=es&nrm=iso

S2306-91552020000100089&lng=es&nrm=iso>. Epub 05-Ene-2020. ISSN 2306-9155.

[7] Castillo Anzures M., Guaña Moya E.J., (2024) “Kanban: Una metodología ágil para la gestión eficiente del flujo de trabajo en el desarrollo de software, una revisión sistemática” [Online] Revista Ingeniería Global. Vol.3 Núm.1, 2024 17-28. [Consultado el 05 de agosto de 2024] Disponible en : <https://doi.org/10.62943/rig.v3n1.2024.68>

[8] Gaete J, Villarroe R, Figueroa I, Cornide-Reyes H, Muñoz R, (2021) “Enfoque de aplicación ágil con Serum, Lean y Kanban” [Online] Ingeniare. Rev. chil. ing. vol.29 no.1 Arica mar. 2021 [Consultado el 05 de agosto de 2024] Disponible en : <http://dx.doi.org/10.4067/S0718-33052021000100141>

[9] IBM. (2020) ¿Qué es un flujo de trabajo? [Internet] IBM [Consultado el 05 de agosto de 2024] Disponible en: <https://www.ibm.com/es-es/topics/workflow>

[10] Torres Calle, Cristian Esteban, Carrión Narváez, Bryam Sebastián, (2024) “Prácticas empíricas en la gestión de proyectos de desarrollo de software a partir de análisis del principio de adaptabilidad” [Online] Trabajo de grado. Universidad Politécnica Salesiana, Ecuador. [Consultado el 05 de agosto de 2024] Disponible en: <https://dspace.ups.edu.ec/handle/123456789/28055>

[11] Merchán-Narváez N J, Palma-Peralta E E, Poma-Japón D X, (2024) “Comparación de metodologías ágiles para el desarrollo de software” [Online] Revista multidisciplinaria arbitrada de investigación científica Vol. 8 Núm. 1 (2024): Edición frecuencia de enero a marzo [Consultado el 05 de agosto de 2024] Disponible en: <https://doi.org/10.56048/MQR20225.8.1.2024.5052-5074>

[12] VERA R A, GOMEZ O, MENDOZA J y PECH J, (2020) “Influencia de la Teoría de Roles en actividades de Gestión: Un experimento controlado con estudiantes de Ingeniería de Software”. RISTI [online]. 2020, n.39 [Consultado 2024-08-22], pp.131-147. Disponible em: http://scielo.pt/scielo.php?script=sci_arttext&pid=S1646-98952020000400010&lng=pt&nrm=iso. ISSN 1646-9895. <https://doi.org/10.17013/risti.39.131-147>.

[13] Flores-Cerna F, Sanhueza-Salazar VM, Valdés-González HM, Reyes-Bozo L, (2022) “Metodologías ágiles: un análisis de los desafíos organizacionales para su implementación” [Online] Vol. 43 No. 1 (2022): January-April 2022 [Consultado el 05 de agosto de 2024] Disponible en: <https://doi.org/10.14483/23448350.18332>

[14] Campos Wright W, Machado Paredes A, (2023) “Redes Neuronales Artificiales para estimar el esfuerzo en proyectos de software. Un estudio comparativo”, [Online] Convención científica internacional, UCIENCIA/23, [Consultado el 05 de agosto de 2024] Disponible en: <https://repositorio.uci.cu/jspui/handle/123456789/10713>

[15] Fernando Villafranca A, Mexicano Santoyo A, Carmona Frausto J C, Dina Valenzuela B, Fernandez Bonilla O D, (2022) “Modelo para la Gestión de

Desarrollo de Software en una Empresa Mediana a Través de Formatos Digitales Basados en Moprosoft” [Online] Artículos del Congreso Internacional de Investigación Academia Journals Oaxaca 2022 [Consultado el 05 de agosto de 2024] Disponible en: https://cdvictoria.tecnm.mx/oferta/maestria_sistemas/Repositorio/2022%20Articulo%20arbitrado%20Modelado.pdf

[16] Núñez-Sánchez Y, González-TorresA, (2020) “Gestión de terceros en el desarrollo de software: propuesta de una metodología (Third-party management in software development: proposal of a methodology) “ [Online] Enfoque UTE Revista Vol. 11 No. 2 (2020) [Consultado el 05 de agosto de 2024] Disponible en: <https://doi.org/10.29019/enfoque.v11n2.621>

[17] Helmut Sy Corvo. (2021)“Minuta Lifeder”, [Online] Humanidades, Administración y economía [Consultado el 05 de agosto de 2024] Disponible en: <https://www.lifeder.com/minuta/>

[18] Peraza-Delgado M, (2022)“Método para la gestión de un proyecto a través de medios digitales en la etapa de pruebas: caso PROMADE” [Online] Revista Nacional de Administración ISSN 1659-4932, Rev. Nac. Adm. vol.13 n.2, 2022 [Consultado el 05 de agosto de 2024] Disponible en: <http://dx.doi.org/10.22458/rna.v13i2.3947>

[19] Pardo-Calvache CJ, Salazar-Escobar GA, Vargas-Arias GF, Masso-Daza JE, (2020) ”Métricas basadas en un enfoque guiado por riesgos para evaluar la comunicación, cooperación y coordinación en equipos de desarrollo global de software” [Online] Rev. Fac. ing. vol.29 no.54 Tunja Jan./Mar. 2020 Epub Dec 30, 2020 [Consultado el 05 de agosto de 2024] Disponible en: <https://doi.org/10.19053/01211129.v29.n54.2020.11759>

[20] Tamayo Espinosa L, Silega Martínez N, (2021)“Gestión de la mantenibilidad desde etapas tempranas en el desarrollo de software” [Online] Rev cuba cienc informat [online]. 2021, vol.15, n.1 [Consultado 2024-08-19], pp. 52-69 . Epub 31-Mar-2021. ISSN 2227-1899. Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992021000100052&lng=es&nrm=iso

Redacción edición.	Revisión y	José Juan Hernández Mora
--------------------	------------	--------------------------



Esta obra está bajo una licencia internacional Creative Commons Atribución 4.0.

TABLA DE ROL DE CONTRIBUCIÓN

Rol	Autor (es)
Conceptualización	Itzel Xochitototl Cote
Curación de datos	José Juan Hernández Mora
Metodología	María Guadalupe Medina Barrera
Administración del proyecto	Elizabeth Cuatecontzi Cuahutle
Recursos	Juan Ramos Ramos
Supervisión	Juan Ramos Ramos
Validación	María Guadalupe Medina Barrera
Redacción (Borrador original)	Itzel Xochitototl Cote