

Gestión y Mejora de Procesos de Desarrollo de Software

Jimmy Gabriel Rivera Ramírez¹

jrivera@upse.edu.ec

<https://orcid.org/0000-0001-8179-6626>

Universidad Estatal Península de Santa Elena
Facultad de Sistema y Telecomunicaciones
La Libertad, Ecuador

Esther Eizabeth Gonzabay De La A

esthergonzabay@upse.edu.ec

<https://orcid.org/0000-0002-3897-3700>

Universidad Estatal Península de Santa Elena
Facultad de Sistemas y Telecomunicaciones
La Libertad, Ecuador

Bolívar Mauricio Mendoza Moran

bmendoza24@alumno.uned.es

<https://orcid.org/0000-0002-7680-7586>

Universidad Nacional de Estudios a Distancia
Facultad de sistemas, Madrid, España

RESUMEN

La mejora de procesos de software (SPI) ha recibido mucha atención tanto en la academia como en la industria. SPI tiene como objetivo mejorar la eficacia del proceso de desarrollo de software. Se han desarrollado varios enfoques diferentes para SPI, incluido el Modelo de madurez de capacidad (CMM) de SEI, más recientemente la Integración del modelo de madurez de capacidad (CMMI). La investigación muestra un informe que indica los beneficios y dificultades de la implantación de la mejora de los SPI en la PYMES. Se hizo una revisión de la literatura y la comparación de varios casos.

Palabras Clave: SPI; procesos; mejora; software; organización.

¹ Autor principal

Correspondencia: jrivera@upse.edu.ec

Management and Improvement of Software Development Processes

ABSTRACT

Software process improvement (SPI) has received much attention in both academia and industry. SPI aims to improve the efficiency of the software development process. Several different approaches to SPI have been developed, including SEI's Capability Maturity Model (CMM), more recently Capability Maturity Model Integration (CMMI). The research shows a report that indicates the benefits and difficulties of implementing the improvement of SPI in SMEs. A review of the literature and the comparison of several cases were made.

Keywords: SPI; processes; improvement; software; organization.

*Artículo recibido 19 agosto 2023
Aceptado para publicación: 28 setiembre 2023*

INTRODUCCIÓN

El desarrollo de software en las empresas, es una de las actividades con menos automatización y planificación según muchos autores, el proceso de desarrollo de software debería ser más mecánico y medible mediante métricas y administrable (Blundell y otros, 1997), por eso es necesario la implantación del SPI (SOFTWARE PROCESS IMPROVEMENT - MEJORA DE PROCESOS DE SOFTWARE) en los departamentos de desarrollo de software de las empresas. El objetivo de este artículo es estudiar las ventajas del uso de las mejoras de procesos en la creación de software.

MÉTODOS

Para el desarrollo de este artículo de revisión se utilizó una metodología de investigación bibliográfica, búsqueda sistematizada de información en bases de datos como Google Scholar y ResearchGate usando cadenas de búsqueda en inglés como: benefits of using the SPI(beneficios del uso del SPI), Resistance factors in projects of the use of SPI (software process improvements) - (Factores de resistencia en proyectos del uso de SPI (mejoras de procesos de software)).

La busque de información dio como resultado treinta fuentes bibliográficas, los cuales fueron leídos y se extrajo la información de experiencia práctica de varios autores en la implantación de las mejoras en los procesos de desarrollos de software.

La búsqueda de información se realizó desde una perspectiva estructurada de:

Beneficios por el uso de SPI

DIFICULTADES EN EL USO DE SPI

Factores de resistencia en proyectos del uso de SPI (mejoras de procesos de software)

Se usó el software MAXQDA, para organizar la información, resaltar los bloques de texto con información clave. Además, MAXQDA permite realizar anotaciones y llevar en un único archivo toda la información organizada respecto al artículo, MAXQDA es una alternativa más económica que su competidor NVIVO.

También se utilizó el software RepetitionDetector2 en versión de prueba de 30 días permite analizar la cantidad de palabras repetidas, lo cuyo resultado se usó para indicar las palabras clave.

DESARROLLO

La mejora del proceso de software (Software process improvement [SPI]) es una iniciativa para evitar la entrega de sistemas de baja calidad (Chevers, 2017). Sin embargo, la concientización y adopción de SPI es baja

La implantación de SPI es costosa e implica un cambio de costumbres y cultura organización, (CHEVERS, 2016) Indica que ha notado la siguiente lista de razones para no adoptar programas de SPI (Chevers, 2017):

Tiempo a consumirse

Falta de recursos

Miedo a hacer cambios SPI y tener que lidiar con una larga curva de aprendizaje

La implantación de SPI provee muchas mejoras y beneficios y también trae algunas dificultades, que se describe a continuación:

(Software Process Improvement [Spi])

(CHEVERS, 2016) Indica que La mejora del proceso de software (Software process improvement [SPI]) es una iniciativa para evitar la entrega de sistemas de baja calidad. Sin embargo, la concientización y adopción de SPI es baja. Las empresas desarrolladoras de software usan programas de SPI reportaron una mejora de la calidad de productos de software como el mayor beneficio. Estos hallazgos confirman la importancia de programas de SPI como medio de producir productos de software de mejor calidad, lo que puede aumentar la probabilidad de que compañías de software consigan contratos globales. Siendo el CMMI un modelo de referencia de prácticas establecidas que se utiliza para evaluar la competencia de los procesos de SI y las metodologías de desarrollo de sistemas estructurados para establecer la repetibilidad de los procesos, lo que, por extensión, puede reducir la dependencia de la dedicación individual para obtener resultados de calidad.

Beneficios por el uso de SPI

Los autores Ferreira e Irigoyen (Ferreiro Ferreira y otros, 2008) indican la siguiente lista de beneficios del uso de SPI:

Beneficios en costos y estimación de horarios

Beneficios en la productividad

Beneficios en la calidad del producto (densidad de defectos)

Beneficios en la satisfacción del cliente

Beneficios financieros

Según (Herbsleb y otros, 1994) indica que después de determinar el costo incurrido en el esfuerzo de SPI, la organización determina la cantidad de dinero ahorrado. Algunos de los métodos más simples utilizados para determinar la cantidad de dinero ahorrado incluyen la cuantificación del valor en dólares de artículos como

Aumento de la productividad.

Detección y corrección temprana de errores.

Reducción total de errores.

Mejora de las tendencias en el trabajo de mantenimiento y garantía.

Eliminación de procesos o pasos de proceso.

Dificultades en el uso de SPI

(CHEVERS, 2016) Manifiesta que la implementación de SPI es costosa y, por lo general, requiere recursos importantes de la empresa.

Siendo la CMM una cultura y nueva forma de pensar para las organizaciones a veces resulta un nuevo impacto a adoptar según (Anabalón, 2005).

Entre las dificultades tenemos:

Problemas en la comunicación de requisitos, comunicación de objetivos y alcances de los procesos de mejora de software. Así como una comunicación interna ineficaz dentro de las organizaciones que intentan la evaluación de sus procesos de mejora según (Anabalón, 2005).

Problemas inherentes a omisiones técnicas de los modelos, específicamente CMM/CMML, respecto de la omisión de aspectos intra culturales de cada organización.

La creencia de que el software se puede mejorar sólo con estándares, métricas y “buenas prácticas”. Pero, si no son bien identificadas y comunicadas pueden derivar irremediablemente en el fracaso de las iniciativas de mejora.

Según (Anabalón, 2005) se dice que CMM/CMMI se basa en pensamientos mecanicistas y que ignora la realidad cultural de cada organización. En otras palabras, se podría inferir, a partir de lo que propone Bill Curran que el modelo CMM/CMMI trata de identificar el desarrollo de software como un proceso ingenieril formal y científico, sin considerar la influencia del software sobre el comportamiento humano.

(Moitra, 2001) afirma que la mayoría de los programas de gestión de cambios se diseñan de manera aislada y no toman en consideración la cultura organizacional y no aseguran la participación de los empleados en el desarrollo del consenso para el cambio deseado, y a su vez se reúnen con el corto plazo insostenible Ganancias

Factores de resistencia en proyectos del uso de SPI (mejoras de procesos de software)

Según (Ahmad, , 2008) se habla de problemas de resistencia de parte de las organizaciones, que dan resultados como objetivos fluctuantes, un pobre manejo de la información distribución de roles no muy clara. Según la investigación de (Brietzke & Rabelo, 2006) los factores de resistencia se categorizan en 2:

Factores del proyecto.

Factores de la organización y

Factores humanos

Los factores organizacionales relacionados con los problemas dentro del alcance de la organización y por lo general están bajo la responsabilidad del administrador principal según se informa y, mientras tanto, los factores del proyecto relacionados con los problemas relacionados con la gestión de proyectos de software, como las actividades de planificación y la distribución de recursos, entre otros.

Factores de proyecto

Hay 4 factores que se categorizan en los factores de proyecto hay 4 factores que se categorizan bajo factores del proyecto a saber:

Presupuesto y estimaciones

Documentación

Calidad

Herramientas y tecnologías.

Presupuesto y estimaciones

Según (Zahran, 1996) uno de los principales obstáculos para la adopción de la mejora de los procesos de software (SPI) es la renuencia que tiene la administración empresarial a invertir en SPI.

Por lo general, esto se debe a la falta de pruebas convincentes del retorno de la inversión (ROI). Esto se debe a la falta general de información confiable sobre los beneficios comerciales de SPI y la ausencia de pruebas contundentes de rendimiento satisfactorio de la inversión (ROI) desde SPI. No todos los ejecutivos corporativos creen en el valor de la mejora del proceso de software. Es el deber de la comunidad de SPI medir, reunir y publicar datos contundentes como evidencia del ROI positivo para SPI. Tales pruebas podrían ayudar a los profesionales del proceso convencer a los empresarios de invertir en SPI.

El presupuesto actual excede la planificación;

La falta de comprensión, por parte de la alta dirección, de que el proyecto de mejora de los procesos de software es un retorno a largo plazo del proceso de inversión;

Falta de visibilidad sobre las actividades de proyectos de mejora continua de procesos de software.

Según (Wiegers, 1996), la falta de progreso en los planes de mejora es frustrante para aquellos que realmente quieren lograr el progreso y esto depuso la importancia del tiempo y los costos en la evaluación del proceso. (OJALA, 2008) describió que el costo es el factor más difícil de manejar en el desarrollo de software. El primer año es el período más difícil para un programa SPI.

Documentación

Según (Ahmad, , 2008) manifiesta que en un proyecto SPI, la documentación es imprescindible para aportar pruebas y diseminaciones en toda la organización de forma formal. Por lo tanto, es útil contar con una infraestructura para la documentación, ya que es una práctica obligatoria en toda la organización.

Según (Beecham & Kuhrmann, 2014) et., la documentación también está ganando importancia en la lista de problemas asociados a SPI. Incluye medición de datos, registro de actas,

coordinación y gestión de la documentación, recopilación de datos del marco operacional que forma las relaciones y dependencias entre lo que se debe hacer, por quién y cómo hacerlo.

Factores clave de resistencia - Documentación

Excesiva documentación y formalidad,

Falta de infraestructura y de manejo de la documentación;

Poca flexibilidad en el uso de la documentación en proyectos de diferentes tipos y tamaños.

Calidad

De acuerdo con (Brietzke & Rabelo, 2006) el factor de calidad se refiere a la persona encargada de la calidad o al grupo de aseguramiento de la calidad en una organización (SQA, PPQA, etc.).

La mayor parte de las veces, el deber principal de esta persona o grupo es garantizar la institucionalización de la mejora de los procesos y, por ello, se sienten directamente la resistencia en la implementación de mejoras.

Según Pires (16), al principio los equipos del proyecto se mostraban reacios a adoptar los nuevos procesos y evaluaciones llevados a cabo por el grupo SQA. Este problema se resolvió involucrando a los altos directivos en el proceso, quienes motivaron a sus equipos a través de conferencias y talleres. Además, el grupo SQA dirigió sus actividades con mayor firmeza hacia el apoyo a la ejecución de las nuevas actividades de proceso.

Factores clave de resistencia calidad

La falta de participación de la alta dirección en la relación entre los equipos del proyecto y la persona o grupo de garantía de calidad;

Falta de tratamiento para garantizar la conformidad del proceso en casos de contratación y/o despido de profesionales cualificados.

Herramientas y tecnología

Los problemas asociados a las herramientas y la tecnología son las segundas preocupaciones más frecuentemente mencionadas por parte de los desarrolladores y directores de proyectos (5). Esto se refiere a la implementación de nuevas tecnologías y herramientas, cantidad de trabajo y presiones que dificultan el uso de nuevas herramientas.

En el caso del proyecto de mejora de procesos de software, la orientación es implementar el uso de herramientas y tecnología en una segunda etapa del proceso.

Factores clave de resistencia - herramientas y tecnología

Automatización de procesos no bien definidos;

Falta de formación sobre las herramientas de apoyo y tecnologías

Presión y ausencia de planificación relativa al período de adaptación

Factores de la organización

Los factores que afectan a implementar SPI son:

Humana

Política

Cultural

Objetivos y Metas

Gestión de cambios

Factores Humanos

Según (Abrahamsson & Iivari, 2002) sin compromiso de todos los niveles organizacionales (humanos) para apoyar a SPI, la iniciativa probablemente fallará o los resultados no se alcanzarán como se esperaba. La experiencia de la alta dirección con un proyecto SPI dará impactos positivos al proceso de mejoramiento. El apoyo a la consulta, como el asesoramiento y la capacitación de los equipos y personal de acción de SPI, es un aspecto crítico para asegurar el éxito del proyecto SPI. (Beecham & Kuhrmann, 2014) afirmaron que las cuestiones organizacionales (especialmente el elemento humano) son factores que contribuyen importantes al éxito de las iniciativas SPI.

(KUMLANDER, 2007), manifiesta que un personal desmotivado es un riesgo desde trabajadores que disminuyen la productividad hasta trabajadores que renuncian a la empresa, en el último caso, la empresa tiene que contratar un nuevo trabajador y capacitarlo al mismo nivel que el empleado que abandono el puesto de trabajo.

Según (Herzberg, 1969) la motivación ha sido un factor importante en la productividad, la calidad y la entrega exitosa de un proyecto dentro del presupuesto y las limitaciones de tiempo.

Según (KUMLANDER, 2007), se debe tomar en cuenta que los desarrolladores expertos, que renuncian a las empresas, crean ciertos problemas para estas empresas, ya que toma varios meses o años para entrenar a una persona que sustituya al programador que dejó su puesto disminuyendo la productividad de la empresa.

Para mantener motivados a los empleados se les puede ofrecer rotación y darle una lista de cargos (posiciones) en la empresa que podría ocupar, como: los testers, los analistas de negocio, los diseñadores del software, los arquitectos de software, etc. El método común aquí es ofrecer al trabajador otra posición diferente de su actual que le resulte interesante.

La posición puede ubicarse en otro lugar ya sea verticalmente, usualmente por encima de la posición actual, como una posición que permite tomar más decisiones, u horizontalmente, lo que significa otra posición en la misma capa organizativa.

Los proyectos de motivación del personal suelen ser los siguientes:

La organización desea alentar a los desarrolladores a aprender algo nuevo que será útil para la organización en el futuro (cercano) proporcionándoles oportunidades para aprender, incluyendo un proyecto dedicado y materiales necesarios;

Los desarrolladores alientan a una organización a trasladarse a otra plataforma demostrando una lista de ventajas y la organización está dispuesta a aceptar esta migración si esas ventajas se muestran durante un proyecto piloto.

Otro factor de resistencia al cambio es la inercia y la experiencia negativa o mala experiencia por parte de los profesionales en software según lo dice (Nathan, 2001).

La inercia y la experiencia negativa

Según (Nathan, 2001) los profesionales de software pueden resistirse a SPI porque no perciben incentivos discernibles para renunciar a prácticas con las que están acostumbrados y sentirse cómodos con SPI. Tal reacción puede no ser necesariamente una respuesta pro-activa a las nuevas prácticas que se están introduciendo, sino más bien una necesidad de continuar con las prácticas actuales y establecidas. Refleja el viejo adagio: "¿por qué arreglar lo que no está roto?". Y según Humphrey (Humphrey, 1998) incluso los profesionales 'inteligentes' no se involucrarán en prácticas que "la lógica, la experiencia y hasta la evidencia sugieren que no deberían.

Algunas de las razones para la inercia según Humphrey (Humphrey, 1998) son:

Una vez que los profesionales aprenden a desarrollar programas que funcionan, también establecen algunas prácticas personales básicas

Estas prácticas personales se arraigan en los más practicantes.

Una mala experiencia anterior de nuevas herramientas y técnicas, no hace que los profesionales piensen que las nuevas prácticas mejorarán su producción

Según (Moitra, 2001) los profesionales del software pueden resistir nuevas prácticas que perciben como una amenaza a su autonomía:

Falta de pruebas de beneficios

Imposición

Restricciones de recursos

Presiones comerciales

Falta de pruebas de beneficios

Los estudios que investigan los factores críticos de éxito de SPI indican que proporcionar a los profesionales pruebas de los beneficios de SPI es un buen motivador (Humphrey, 1998). Por otra parte, los profesionales del software pueden mostrar intransigencia hacia nuevas y mejores prácticas que no necesariamente se mejorarán proporcionándoles pruebas. Humphrey argumenta que los profesionales no usarán nuevas prácticas aun cuando existan pruebas claras de que estas prácticas o métodos ayudan.

Imposición

De acuerdo con (Hantos & Gisbert, 2000) los programadores de SPI que se inician desde el nivel corporativo a menudo enfrentan barreras de los practicantes de software. Esto se debe a que los practicantes de software resisten las iniciativas que perciben como impuestas a ellos.

Factores clave de resistencia - humana

Falta de compromiso en todos los niveles de las organizaciones

Falta de adherencia y participación de todo el individuo involucrado en proyectos SPI

Carencia de profesionales experiencia y destreza

Falta de liderazgo y respaldo por nivel de alta dirección

Política

Según (Wheeler & Duggins:, 1998) el factor de política de calidad es importante a la construcción para un departamento de Calidad de Software SQA. El establecimiento de una política de calidad, que es una de las cuestiones políticas, viene después del compromiso de la alta dirección. Las políticas y estándares de calidad para los esfuerzos de SPI describen las metas y objetivos organizacionales relacionados con la calidad.

Factores clave de resistencia

Falta de establecimiento de políticas organizacionales

Falta de establecimiento de una política de calidad

Cultural

(Taylor & McGraw, 2005) demostró que, para asegurar éxito en un programa cultural del cambio, un líder deber construir, desplegar, conducir, y poseer cada iniciativa que va adelante debe ser decidida correctamente. Sin embargo, cada programa de cambio cultural requiere una buena cooperación tanto del personal directivo como del técnico táctico; los programas de mejora fallarán si cualquiera de los dos grupos se deja fuera o se subraye.

Factores clave de resistencia - política

Falta de experiencia en la implementación de cambios culturales.

Objetivos y Metas del uso de SPI

Según la investigación de (Wiegers, 1996), si las metas, los plazos, y los resultados esperados por los encargados son imprácticos, el esfuerzo hacia SPI puede ser un fracaso. Según lo mencionado por (Lazic, 2008) el desarrollo de software se mide en términos de progreso general en la reunión de objetivos funcionales y de negocio que aseguran cómo el proceso de software se está haciendo bien. Mientras tanto, (Clarke & Osterweil, 2000) encontraron una razón por la cual el proceso de software necesita tener procesos indefinidamente en curso. Es esencial que se especifiquen los objetivos claros antes, de manera que el progreso hacia esos objetivos pueda ser continuamente monitoreado, y para que las revisiones de los objetivos, de los procesos o de ambas se puedan hacer persistentemente.

Gestión de administración

Es necesario realizar un análisis inicial para determinar si la iniciativa SPI es apta para los objetivos e intereses de la organización. Esto también es discutido y apoyado para que el equipo de SPI se utilice para facilitar activamente los esfuerzos hacia los cambios por parte de los equipos del proyecto en lugar de simplemente comprobar la situación del proceso en curso para reportar una larga y deprimente lista de hallazgos (Wiegers, 1996). (Miler1 & Górski , 2004) destacó que, para tener un éxito en el proceso de mejoras de software, el establecimiento de riesgo y gestión de cambios que no está explícitamente definido conducirá a un proceso de fallido.

Según (Moitra, 2001) cualquier programa de gestión de cambios debe ser tratado como un proyecto, con un enfoque exhaustivo de gestión de riesgos. Se ha visto que agentes de cambio muestran incapacidad para establecer la credibilidad del programa de cambio, es bien sabido que ha contribuido al fracaso de muchas iniciativas de mejoramiento de procesos. Se necesita un enfoque estructurado basado en la administración de proyectos, con criterios e hitos de entrada y salida bien definidos para cada fase de las iniciativas de mejoramiento de procesos.

Factores clave de resistencia – gestión de administración

Enfoque simultáneo en muchas áreas de mejora.

Evaluación insuficiente e ineficaz del proceso de software actual

Enfoque simultáneo en muchas áreas de mejora

Otros factores de resistencia

Visión pobre

Enfoque a corto plazo.

Falta de sentido de urgencia.

Búsqueda de la Certificación

Falta de comunicación efectiva.

Visión pobre

Para (Moitra, 2001) En muchas organizaciones los programas de gestión del cambio se conciben sin tener una visión bien pensada y una clara comprensión del contexto. Muchas organizaciones han adoptado modelos de mejora de procesos de software sin tener una comprensión de cómo

encaja en su entorno de negocios y qué beneficios pueden obtener de él a largo plazo, y por qué se esfuerzan por lograr un nivel de SEI-CMM en particular.

Enfoque a corto plazo.

(Moitra, 2001) indica que la mayoría de los programas de gestión de cambios orientados a la mejora de los procesos suelen tener un enfoque a corto plazo, no integran la unidad de administración de cambios con la estrategia y los objetivos de la organización a largo plazo.

Falta de sentido de urgencia.

(Moitra, 2001) afirma que rara vez tiene un programa de cambio, dirigido a la mejora del proceso de software, ha tenido éxito sin un sentido de urgencia. La incapacidad para reforzar e instituir cambios en el proceso (mantener cambios) es otro factor que a menudo conduce al fracaso en las unidades de administración de cambios organizacionales. En la mayoría de los casos, no hay una sucesión rápida de resultados, sino sólo una serie de iniciativas de cambio, o un nuevo “Mantra”, que conducen a que los empleados no estén convencidos.

Gestión de la resistencia

Para (Stelzer & Mellis, 1999) la mayoría de los agentes de cambio enfrentan una fuerte resistencia del personal de línea y, por lo tanto, es importante que superen esta resistencia. Desafortunadamente, la mayoría de los agentes de cambio no entienden la anatomía de la resistencia y eventualmente no tienen éxito en sus esfuerzos. Se ha encontrado que la resistencia al cambio se debe principalmente a una percepción de:

La incertidumbre y el escepticismo sobre la efectividad de los nuevos procesos y los posibles beneficios de ellos;

Pérdida de control y/o prominencia dentro de la organización como resultado de los procesos nuevos/mejorados;

Aumento de la sobrecarga de trabajo y aumento de la demanda de tiempo.

La calidad de los profesionales del proceso.

Para (Moitra, 2001) Las personas que se ocupan de la garantía de calidad y la mejora del proceso a menudo se ven como un obstáculo en lugar de ayudar. La razón principal es que la calidad y el

proceso de mejora de las personas son a menudo bastante teóricos, ellos mismos no entienden muy bien los procesos existentes de desarrollo de software y el contexto en el que se utilizan.

Por lo tanto, su enfoque hacia la mejora de procesos no aborda las necesidades reales de los desarrolladores y el entorno empresarial. También se ha notado que los agentes de cambio responsables de las mejoras del proceso de software a menudo no tienen una experiencia de desarrollo significativa, y como resultado su enfoque hacia la mejora de procesos se vuelve irreal y poco práctico (Moitra, 2001)

Otra observación es que a veces la mejora del proceso las personas se vuelven demasiados entusiastas acerca de un modelo de proceso o paradigma de mejora de procesos, sin entender y ser capaz de explicar cómo va a ayudar y de qué manera. Una observación que he tenido recientemente es con respecto a la implementación de PSP (personal Software Process), donde el campeón de mejora de procesos estaba abogando demasiado duro para el uso de PSP, pero él mismo no entendía bien PSP y no tenía experiencia con él.

A veces los agentes de cambio tienden a impulsar la implementación de un modelo/técnica de mejora de procesos no lo adaptan la cultura organizacional y empresarial, lo que asegura que los resultados no sigan.

DISCUSION

Dentro de la implantación de SPI se ha notado que el capital humano de la empresa y su cultura organización juegan un papel fundamental en la adopción y puesta en marcha del proceso de mejora de software.

Se hace necesario la existencia de un líder más conocido como gestor de cambio, pero existen dificultades para poder implantar el SPI

El personal de gestor de cambio puede ser una persona muy teórica y podría no puede motivar adecuadamente al personal de sistemas.

El personal de sistemas puede considerar innecesario cambiar la forma de hacer los procesos, ya que los sistemas actuales funcionan bien y posiblemente no hubo comunicación efectiva para poner en su conocimiento el beneficio futuro como la reducción de tiempos en el mantenimiento del software y la reprogramación.

Se debe alinear los objetivos a corto plazo del proceso de SPI con los objetivos de la empresa.

Se debe hacer conocer a los profesionales del software los beneficios a corto y largo plazo del SPI, entre ellos está la detección y corrección temprana de errores algo que los programadores saben que es un dolor de cabeza cuando aparece un error y deben encontrar una solución contra el tiempo.

Debe haber una comunicación efectiva y el gestor de cambio debe estar constantemente con los profesionales de software durante el desarrollo del proceso SPI.

CONCLUSIONES

Ya que la informática está ganando gran importancia, viene a ser software que ayuda en los procesos de toda empresa, pues el desarrollo de software debe dejar de ser desarrollo artesanal y poco medible para convertirse en desarrollos de software planificados en el tiempo, sistematizados, con procesos de control de calidad (QC) a todo nivel para aseguramiento de la calidad (QA). Es importante que la gerencia sea quien dirige la implantación de procesos de mejora de desarrollo de software y que tenga en cuenta los objetivos a corto, mediano y largo plazo, siendo uno de los principales objetivos un producto de software funcional, desarrollado dentro de los tiempos estipulados, y este software tenga la menor cantidad de errores (issues) para no tener que reprogramar el software, y esto se logrará con la aplicación de SPI en todas las etapas de desarrollo de software.

REFERENCIAS BIBLIOGRAFÍA

Abrahamsson , P., & Iivari, N. (03 de 02 de 2002). *RESEARCHGATE*. Commitment in Software

Process Improvement - In Search of the Process:
https://www.researchgate.net/publication/232652611_Commitment_in_Software_Process_Improvement--In_Search_of_the_Process_PDF

Ahmad, R. (22 de 06 de 2008). *researchgate*. Issues in the implementation of software process

improvement project in Malaysia:
https://www.researchgate.net/publication/228737072_Issues_in_the_implementation_of_software_process_improvement_project_in_Malaysia

- Ahmad, R., Nizam, M., & Mohd, H. (23 de 09 de 2008). *Resistance factors in the implementation of software process improvement project*. Conference: Information Technology: https://www.researchgate.net/publication/4376419_Resistance_factors_in_the_implementation_of_software_process_improvement_project
- Anabalón, J. R. (4 de noviembre de 2005). *Las Causas más Comunes de Falla en la Implantación de Mejoras en Software*. https://www.researchgate.net/publication/324919028_Las_Causas_mas_Comunes_de_Falla_en_la_Implantacion_de_Mejoras_en_Software
- Asatiani, A., Penttinen, E., & Kumar, A. (01 de 02 de 2019). *Uncovering the nature of the relationship between outsourcing motivations and the degree of outsourcing: An empirical study on Finnish small and medium-sized enterprises*. <https://doi.org/10.1177/0268396218816255>
- BASTOS TIGRE, P., & SILVEIRA MARQUES, F. (2009). *Desafíos y oportunidades de la industria del software en América Latina* (Vol. 1). Bogotá, Colombia: COMISIÓN ECONÓMICA PARA AMÉRICA LATINA Y EL CARIBE. https://www.cepal.org/sites/default/files/publication/files/1989/S33826D4412009_es.pdf
- Beecham, S., & Kuhrmann, M. (mayo de 2014). *Artifact-based software process improvement and management: a method proposal*. Proceedings of the 2014 International Conference on Software and System Process: <https://doi.org/10.1145/2600821.2600839>
- Blundell, J., Hines, M., & Stach, J. (1997). *The measurement of software design quality*. Annals of Software: <https://doi.org/10.1023/A:1018914711050>
- Brietzke, J., & Rabelo, A. (12 de 10 de 2006). *Resistance Factors in Software Process Improvement*. CLEI ELECTRONIC JOURNAL, VOLUME 9, NUMBER 1, PAPER 4, JUNE 2006: <http://www2.clei.org/cleiej/papers/v9i1p4.pdf>
- CHEVERS, D. (3 de 10 de 2016). *SOFTWARE PROCESS IMPROVEMENT: AWARENESS, USE, AND BENEFITS IN CANADIAN SOFTWARE DEVELOPMENT FIRMS*.

- <http://dx.doi.org/10.1590/s0034-759020170206>
- http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0034-75902017000200170
- Chevers, D. (MARZO de 2017). *Software process improvement: Awareness, use, and benefits in Canadian software development firms*. *Revista de Administração de Empresas* : DOI: <http://dx.doi.org/10.1590/S0034-759020170206>
- Clarke, L., & Osterweil, L. (2000). Continuous Self-Evaluation for the Self-Improvement of Software. *International workshop on Self-adaptive software*, 27- 39.
- Ferreiro Ferreira, A., Santos, G., Cerqueira, R., Montoni, M., Barreto, A., Rocha, A., . . . Silva Filho, R. (24 de 07 de 2008). *ROI of Software ProcessImprovement at BLInformática: SPIdex is Really Worth it*. SOFTWARE PROCESS IMPROVEMENT AND PRACTICE: <https://onlinelibrary-wiley-com.ezproxy.uned.es/doi/epdf/10.1002/spip.392>
- Hantos, P., & Gisbert, M. (Enero de 2000). *Identifying software productivity improvement approaches and risks: Construction industry case study*. IEEE Software; Los Alamitos: <https://search-proquest-com.ezproxy.uned.es/docview/215830542?pq-origsite=summon>
- Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., & Zubrow, D. (1 de agosto de 1994). *Software Engineering Institute*. Benefits of CMM-Based Software Process Improvement: Initial Results: https://resources.sei.cmu.edu/asset_files/TechnicalReport/1994_005_001_16310.pdf
- Herzberg, F. (20 de 06 de 1969). *semanticscholar*. One More Time: How Do You Motivate Employees.: <https://pdfs.semanticscholar.org/ca2a/a2ae02ac5b738b55b12b7324fac59571b1c1.pdf>
- Humphrey, W. (02 de marzo de 1998). *Why don't they practice what we preach?* *Annals of Software Engineering*: <https://link.springer.com/article/10.1023%2FA%3A1018997029222>
- Kotter, J. (2 de 05 de 1996). *Leading Change*. Nova Southeastern University: https://ac.els-cdn.com/S0090261697900498/1-s2.0-S0090261697900498-main.pdf?_tid=03bcf073-3cdb-4d74-b56c-82a5e84fa732&acdnat=1543209816_6e24769a82abdeaa5011dda0405f6963

- KUMLANDER, D. (16 de 02 de 2007). *WSEAS Conferences*. Personnel motivating projects: reasons, implementation and risks : <http://www.wseas.us/e-library/conferences/2007corfu/papers/540-211.pdf>
- Lazic, L. a. (2008). Cost Effective Software Test Metrics. *WSEAS Transactions on Computers*,, 559-619 Vol 6, No.7, pp. .
- Miler1, J., & Górski , J. (11 de 11 de 2004). *European Software Process Improvement Conference EuroSPI'2004*. Risk-driven Software Process Improvement - a Case Study : <https://pdfs.semanticscholar.org/e1af/d1d105b930348caa7b06e36e5b4ea72714dd.pdf>
- Moitra, D. (1 de Marzo de 2001). *Managing change for software process improvement initiatives: a practical experience-based approach*. onlinelibrary-wiley-com: <https://onlinelibrary-wiley-com.ezproxy.uned.es>
- Nathan, B. (01 de 12 de 2001). *Motivators and de-motivators in software process improvement : an empirical study*. University of Hertfordshire PhD Theses Collection: <https://uhra.herts.ac.uk/handle/2299/14050>
- OJALA, P. (23 de 11 de 2008). *Transactions on Information Science & Spplications*. Experiences of Implementing a Value-Based Approach: https://www.academia.edu/34983253/Experiences_of_Implementing_a_Value-Based_Approach
- Prenger, K. D. (1 de diciembre de 1997). *The Defense Technical Information Center (DTIC®)*. The Defense Technical Information Center (DTIC®): <http://www.dtic.mil/dtic/tr/fulltext/u2/a346053.pdf>
- Stelzer, D., & Mellis, W. (04 de 05 de 1999). *Success Factors of Organizational Change in Software Process Improvement*. (Universitaet zu Koeln, Germany) Software Process Improvement and Practice, Volume 4, Issue 4 Copyright □ 1999 John Wiley & Sons Ltd: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.39.3087>
- Taylor, D., & McGraw, G. (23 de mayo de 2005). *Adopting a Software Security Improvement Program*. <https://publications.computer.org/security-and-privacy/>: <https://www.cigital.com/papers/download/bsi8-program.pdf>

Wheeler, S., & Duggins, S. (1998). Improving software quality. *ACM Southeast Regional Conference*, 300-309.

Wiegers, K. (12 de enero de 1996). <http://citeseerx.ist.psu.edu>. Software Process Improvement: Ten Traps to Avoid: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.41.5452&rep=rep1&type=pdf>

Zahran, S. (20 de enero de 1996). *Business and cost justification of software process improvement "ROI from SPI"*. researchgate.net: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.6063&rep=rep1&type=pdf>