



**FACULTAD DE INGENIERÍA, ARQUITECTURA Y
URBANISMO
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS
TESIS**

**DESARROLLO DE UN MODELO DE PROCESOS
PARA LA CONSTRUCCIÓN DE SOFTWARE EN
MICROEMPRESAS PERUANAS
DESARROLLADORAS DE SOFTWARE. CASO DE
ESTUDIO: SISTEMA INTELIGENTE ERP SAC
PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO
DE SISTEMAS**

Autor (es):

Bach. Huancas Montenegro Jeiner

ORCID: <https://orcid.org/0000-0003-2353-8164>

Bach. Vargas Moreno Jorge Luis

ORCID: <https://orcid.org/0000-0003-3964-2919>

Asesor:

Mg. Atalaya Urrutia Carlos William

ORCID: <https://orcid.org/0000-0002-2761-4868>

Línea de Investigación:

Infraestructura, Tecnología y Medio Ambiente

Pimentel - Perú

2021

DESARROLLO DE UN MODELO DE PROCESOS PARA LA
CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS
PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE
ESTUDIO: SISTEMA INTELIGENTE ERP SAC.

Aprobación del Informe

Bach. Huancas Montenegro Jeiner

Autor

Bach. Vargas Moreno Jorge Luis

Autor

Mg. Atalaya Urrutia Carlos William

Asesor

Dr. Vásquez Leyva Oliver

Presidente

Mg. Atalaya Urrutia Carlos William

Secretario

Mg. Díaz Vidarte Miguel Orlando

Vocal

DEDICATORIA

La presente investigación está dedicada a mis padres, Pedro Huancas Rodriguez y Dermi Imelda Montenegro Bravo, así como también, a mis hermanos; gracias por haber sido mi apoyo en toda mi carrera universitaria a lo largo de mi vida; también a aquellas personas especiales que me acompañaron en esta etapa, aportando en mi formación tanto profesional como ser humano.

JHM

La presente investigación está dedicada a mis padres, Alejandro Vargas Garamendi, y Aleja Moreno Gomez, así como a mi esposa Lourdes Carol Arias Ramos y, a mi hermoso hijo Stefano Misael Vargas Arias y a toda mi familia; gracias por haber sido mi apoyo en toda mi carrera universitaria a lo largo de mi vida; también a aquellas personas especiales que me acompañaron en esta etapa, aportando mi formación tanto profesional como ser

humano.

JLVM

AGRADECIMIENTO

Agradecemos en primer lugar a Dios por bendecirnos en la vida, por guiarnos a lo largo de nuestra existencia, por ser nuestro apoyo y fortaleza en aquellos momentos de dificultad y de debilidad. Por su inmensa gratitud que siempre llena nuestras vidas.

Agradecemos a nuestras familias, por ser el pilar fundamental y habernos apoyado incondicionalmente, pese a las adversidades e inconvenientes que se presentaron y siempre estuvieron a nuestro lado.

Agradecemos a la Universidad Señor de Sipán, casa de estudios lambayecana, la cual nos dio la bienvenida al mundo laboral como tal, por las oportunidades brindadas las cuales son incomparables y que, ante la actual coyuntura, ha brindado apoyo y solidaridad que, en algún momento se pensó no se podrían dar para la finalización de esta carrera profesional.

Agradecemos, de la más sincera manera, al Mg. Heber Ivan Mejia Cabrera quien ha brindado su apoyo, asesoría y capacitación para poder desarrollar de la mejor manera esté presente informe de investigación. Verdaderamente que, se ha aprendido muchísimo de usted. Todas las enseñanzas de usted en esta parte final de la carrera, son uno de los recuerdos más hermosos que se tiene del largo trayecto en esta casa de estudios.

Agradecemos, finalmente, a la empresa Sistema Inteligente ERP SAC por haber permitido el uso y recopilación de información para el presente informe de investigación, lo que ha llevado a lograr la consecución del presente objetivo académico.

RESUMEN

Esta investigación se centra en el desarrollo de un modelo de procesos para la construcción de software en microempresas peruanas. Ello, ante la problemática con la que cuentan actualmente estas microempresas, quienes carecen de modelos de procesos para la construcción de software, los cuales les sirvan de guía, para el desarrollo de productos con altos estándares de calidad, o que, por ejemplo, sí se halle la existencia de estándares, sin embargo, que estos se encuentren enfocados en el desarrollo de software para organizaciones de mucha mayor envergadura, más no para las microempresas. Por ello, esta investigación tiene como objetivo principal desarrollar un modelo de procesos para la construcción de software en las microempresas desarrolladoras de software en el Perú. Para llevar a cabo dicho propósito, se caracterizaron los modelos de procesos para la construcción de software, luego se procedieron a analizar los modelos principales enfocados en la microempresa, y que se pudieron distinguir mediante una revisión bibliográfica de papers, posteriormente se diseñó el modelo de procesos de construcción de software adecuado a una microempresa desarrolladora de software como caso de estudio, y basado en la Norma ISO/IEC 29110 y el Modelo MoProSoft, y finalmente, se procedió a validar mediante juicio de expertos el modelo propuesto. Los resultados obtenidos revelaron un modelo dividido en dos (02) procesos principales, la Gestión de Proyectos y la Implementación de Software, ambos basados en la Norma ISO/IEC 29110 y el Modelo MoProSoft, los cuales a su vez se desplegaban en diez (10) actividades, teniendo como soporte para el modelado a la herramienta de código abierto ArchiMate. Se concluye que, gracias a este modelo propuesto se mostró una mejora en cuanto al nivel de las dimensiones costo de calidad por proyecto, defectos por proyecto, rentabilidad del proyecto y satisfacción del cliente, de un grupo de proyectos que han sido desarrollados por la empresa caso de estudio.

Palabras Clave: Desarrollo de software, desarrollo de software en micro y pequeñas empresas, calidad de procesos de software, mypes productores de software, calidad de software.

ABSTRACT

This research focuses on the development of a process model for the construction of software in Peruvian microenterprises. This, given the problems that these micro-companies currently have, who lack process models for the construction of software, which serve as a guide, for the development of products with high quality standards, or who, for example, do not have the existence of standards, however, that these are focused on the development of software for organizations of much larger scope, but not for micro-enterprises. For this reason, the main objective of this research is to develop a process model for the construction of software in micro-companies that develop software in Peru. To carry out this purpose, the process models for the construction of software were characterized, then the main models focused on microenterprise were analyzed, and that could be distinguished through a bibliographic review of papers, later the model of software construction processes suitable for a software development micro-enterprise as a case study, and based on the ISO / IEC 29110 Standard and the MoProSoft Model, and finally, the proposed model was validated through expert judgment. The results obtained revealed a model divided into two (02) main processes, Project Management and Software Implementation, both based on the ISO / IEC 29110 Standard and the MoProSoft Model, which in turn were deployed in ten (10) activities, having the ArchiMate open source tool as support for modeling. It is concluded that, thanks to this proposed model, an improvement was shown in terms of the level of the dimensions cost of quality per project, defects per project, profitability of the project and customer satisfaction, of a group of projects that have been developed by the company case study.

Keyword: Software development, Software development micro and small business, Software process quality, software-producing mypes, software quality.

ÍNDICE

I.	INTRODUCCIÓN	13
1.1.	Realidad Problemática.	13
1.2.	Antecedentes de Estudio.	22
1.3.	Teorías relacionadas al tema.	34
1.4.	Formulación del Problema.	57
1.5.	Justificación e importancia del estudio.	57
1.6.	Hipótesis.	59
1.7.	Objetivos.	59
1.7.1.	Objetivo general.....	59
1.7.2.	Objetivos específicos.....	59
II.	MÉTODO	61
2.1.	Tipo y Diseño de Investigación.	61
2.2.	Población y muestra.....	61
2.3.	Variables, Operacionalización.....	63
2.4.	Técnicas e instrumentos de recolección de datos, validez y confiabilidad. 66	
2.5.	Procedimiento de análisis de datos.....	70
2.6.	Criterios éticos.	76
2.7.	Criterios de Rigor Científico.	77
III.	RESULTADOS	79
3.1.	Resultados en tablas y figuras.	79
3.2.	Discusión de resultados.	86
3.3.	Aporte práctico.....	88
IV.	CONCLUSIONES Y RECOMENDACIONES.....	144
3.4.	Conclusiones.....	144

3.5. Recomendaciones.....	145
REFERENCIAS.....	146
ANEXOS	155

ÍNDICE DE ANEXOS

Anexo 1. Resolución de aprobación del proyecto de investigación.....	155
Anexo 2. Carta de presentación del estudiante para realizar caso de estudio... ..	156
Anexo 3. Carta de aceptación de la institución para la recolección de datos.	157
Anexo 4. Instrumento de recolección de datos - Ficha de Observación.....	158
Anexo 5. Instrumento de recolección de datos - Formulario Google.....	159
Anexo 6. Formatos empleados en el desarrollo del modelo.....	160
Anexo 7. Herramientas empleadas para el Proceso de Gestión de Proyectos ..	165
Anexo 8. Herramientas empleadas para el Proceso de Implementación de Proyectos	167
Anexo 9. Herramienta 29110-in-a-box para el desarrollo de los formatos	168
Anexo 10. Acta de constitución de proyecto.....	170
Anexo 11. Formato de Acta de Reunión	174
Anexo 12. Acta de Aceptación	175
Anexo 13. Formato de registro de trazabilidad.....	176
Anexo 14. Matriz de consistencia.....	177
Anexo 15. Ficha para Juicio de Expertos.	178
Anexo 16. Perfiles profesionales de los expertos evaluadores.	180
Anexo 17. Comunicación con los expertos de evaluación.	186
Anexo 18. Juicio de Expertos.....	189
Anexo 19. Diapositivas para capacitación del modelo propuesto.....	195
Anexo 20. Capacitación del Modelo en Sistema Inteligente ERP SAC	196
Anexo 21. Proyectos evaluados Pres Test.....	198
Anexo 22. Proyectos evaluados Post Test.....	202
Anexo 23. Cronogramas en MS-Project de proyectos desarrollados	205
Anexo 24. Evidencias fotográficas	207

ÍNDICE DE FIGURAS

Figura 1, Áreas de costo relacionadas con la deficiente calidad de software en EEUU. _____	16
Figura 2, Porcentaje de empresas que usan modelos de procesos. _____	21
Figura 3, Evolución del método de aseguramiento de la calidad y cambio en el paradigma de la calidad. _____	37
Figura 4, Pasos para adopción del software perfil básico ISO/IEC29110. _____	47
Figura 5, Procesos del ciclo de la vida del Software ISO12207. _____	52
Figura 6, Niveles de madurez de CMMI. _____	54
Figura 7, Modelo para la construcción de software basado en CMMI-DEV _____	55
Figura 8, Categorías y procesos de MoProSoft. _____	56
Figura 9, Modelo para la construcción de software basado en el Modelo Competisoft _____	57
Figura 10, Encuesta desarrollada vía Formulario Google _____	68
Figura 11, Línea de tiempo para la evaluación de los proyectos _____	76
Figura 12, Indicador Nivel de Claridad del Modelo _____	79
Figura 13, Indicador Nivel de Objetividad del Modelo _____	79
Figura 14, Indicador Nivel de Actualidad del Modelo _____	80
Figura 15, Indicador Nivel de Organización del Modelo _____	80
Figura 16, Indicador Nivel de Suficiencia del Modelo _____	81
Figura 17, Indicador Nivel de Intencionalidad del Modelo _____	81
Figura 18, Indicador Nivel de Consistencia del Modelo _____	82
Figura 19, Indicador Nivel de Coherencia del Modelo _____	82
Figura 20, Indicador Nivel de Metodología del Modelo _____	83
Figura 21, Indicador Nivel de Pertinencia del Modelo _____	83
Figura 22, Indicador Defectos por proyecto. _____	84
Figura 23, Cantidad promedio de defectos por proyecto _____	85
Figura 24, Indicador Satisfacción del Cliente _____	86
Figura 25, Estructura de la calidad de software. _____	89
Figura 26, Modelo de procesos ISO/IEC 29110 _____	100
Figura 27, Proceso de Gestión de Proyectos. _____	103
Figura 28, Proceso de Implementación de Software. _____	104

Figura 29, Categorías y Procesos de MoProSoft. _____	112
Figura 30, Logo Sistema Inteligente ERP SAC. _____	119
Figura 31, Organigrama empresarial Sistema Inteligente ERP SAC. _____	120
Figura 32, Plan de Procesos Sistema Inteligente ERP SAC. _____	120
Figura 33, Modelo de construcción de software propuesto. _____	121
Figura 34, Actividades para el desarrollo del Modelo. _____	122
Figura 35, Procesos del Perfil Básico ISO/IEC 29110. _____	122
Figura 36, Procesos del Perfil Básico MoProSoft. _____	123
Figura 37, Vista del Proceso de Gestión de Proyectos. _____	126
Figura 38, Objetivos de la vista del Proceso de Gestión de Proyectos. _____	127
Figura 39, Actividad PGP01 Planificación del Proyecto _____	128
Figura 40, Actividad PGP02 Ejecución del Plan del Proyecto _____	129
Figura 41, Actividad PGP03 Evaluación y Control del Proyecto _____	130
Figura 42, Actividad PGP04 Cierre del Proyecto _____	131
Figura 43, Vista del Proceso de Implementación de Software _____	134
Figura 44, Objetivos de la vista del Proceso de Implementación del Software _____	135
Figura 45, PIS01 Iniciación y Análisis de Requisitos de Software _____	136
Figura 46, PIS02 Arquitectura, Diseño y Construcción de Software _____	137
Figura 47, PIS03 Integración y Pruebas de Software _____	138
Figura 48, PIS04 Entrega del Producto _____	139
Figura 49, Evaluación general del modelo de procesos _____	143

ÍNDICE DE TABLAS

Tabla 1. Población de estudio. _____	62
Tabla 2. Muestra de estudio. _____	62
Tabla 3. Sub grupo de la muestra. _____	63
Tabla 4. Operacionalización de la variable independiente _____	64
Tabla 5. Operacionalización de la variable dependiente _____	65
Tabla 6. Indicador Satisfacción del Cliente _____	85
Tabla 7. Fuentes de información. _____	90
Tabla 8. Cadenas de búsqueda para cada base de datos. _____	90
Tabla 9. Papers relacionados con la usanza de modelos/estándares de construcción de software. _____	91
Tabla 10. Modelos de procesos de construcción de software. _____	94
Tabla 11. Tabla resumen de modelos de procesos de construcción de software.	95
Tabla 12. Beneficios Día-a-Día y beneficios comerciales. _____	97
Tabla 13. Descripción de los perfiles ISO/IEC 29110. _____	99
Tabla 14. Deployment Package disponibles en ISO/IEC 29110 _____	102
Tabla 15. Descripción de Roles en MoProSoft _____	113
Tabla 16. Equivalencia entre procesos involucrados en el mapeo. _____	116
Tabla 17. Objetivos del proceso de Gestión de Proyectos del Perfil Básico ISO/IEC 29110. _____	124
Tabla 18. Objetivos del proceso de Implementación de Software del Perfil Básico ISO/IEC 29110. _____	132
Tabla 19. Expertos para validación de instrumento. _____	140
Tabla 20. Aspectos de validación del juicio de expertos _____	140
Tabla 21. Resultados validación de expertos _____	142
Tabla 22. Promedio de validación del modelo _____	142

I. INTRODUCCIÓN

1.1. Realidad Problemática.

Las Micro y Pequeñas Empresas, denominada PYME son aquella entidad de carácter económico, establecida por una persona de tipo, bien sea de carácter jurídico o natural, sin importar la condición de organización por la cual han sido originadas, la cual presenta un fin, el de lograr el desarrollo de distintas actividades, ya sean de tipo extractivas, de transformación, de prestación de servicios, de producción o de comercialización de bienes (SUNAT, 2019).

Según Al-Herwi (2019), al referirse a las PYME se hace alusión a aquellas pequeñas y medianas empresas, llamadas en el contexto internacional como Small and medium-sized enterprises (SME), las cuales son empresas cuyo número de colaboradores cae por debajo de ciertos límites de acuerdo a los distintos países y sus legislaciones. El conjunto que conforman las PYMES, lleva a cabo una labor sumamente relevante en las economías modernas de cada nación, justificando así, ser la más interesante e innovadora fuerza económica que puede existir en estas naciones. El número de colaboradores con los que cuentan las PYMES también tiene una variación entre una industria u otra, por lo que, no existe un estándar establecido por cada industria o sector.

El término PYME hace referencia a aquellas empresas que se encuentran dentro de una categoría que, básicamente, no corresponde a alguna gran organización. Este término carece de un concepto mundialmente aceptado y estandarizado, por lo que, engloba un amplio repertorio de títulos, de enunciados y de definiciones, los cuales pueden ser adecuados libremente en distintos países de una manera diversa, no siendo ajenas aquellas instituciones las cuales informan sobre el acontecer y sobre las estadísticas de este sector que incluye a las micro y pequeñas empresas constructoras de software (Sánchez, Colomo, de Amescua, & O'Connor, 2017). Entre aquellos criterios comúnmente utilizados para lograr conceptualizar a este sector de empresas, se tiene al número de los colaboradores que laboran en la empresa, a los montos de ventas totales, al nivel de la inversión por parte de los accionistas, al monto de capital inicial, al total de activos netos, etc. No obstante, la base de

definición más común utilizada es el empleo, y aquí nuevamente, hay una cierta diferencia en la conceptualización en lo que respecta a los límites de tamaño inferior o superior de las PYMES. Según la Comisión Europea de la Unión Europea (2014), define que una micro empresa tienen un rango de 10 trabajadores a menos y una pequeña empresa tiene un rango de hasta 50 trabajadores.

En el Perú, el Instituto Nacional de Estadística e Informática (INEI) junto con el Ministerio de la Producción - PRODUCE, prepararon el documento “Perú: Estructura Comercial 2018”, el cual reveló que el 94.9% corresponde a microempresas, así clasificadas por el monto de ventas que no excede de 622,500.00 soles por año, que es equivalente a 150 Unidades Tributarias (UIT). El 4,2% corresponde al segmento de pequeñas empresas que venden entre 150 y 1.700 UIT, mientras que el 0.6% corresponde a las grandes y medianas empresas que tienen ventas superiores a 1.700 UIT. Finalmente, solo el 0.3% restante corresponde a entidades públicas (INEI, 2019).

Las medianas y pequeñas empresas (PYME) representan una fuente muy trascendental de empleo y crecimiento del tipo económico en aquellas naciones que conforman una parte en la Organización para la Cooperación y el Desarrollo Económico (OCDE). Bajo la perspectiva y enfoque de esta organización internacional con sede en París, Francia, el conjunto constituido por la totalidad de PYMES es considerado como la representación que personifica y domina a casi el total de organizaciones de tipo empresarial, sobre todo ya que, las mismas se encuentran representando una cantidad porcentual que fluctúa alrededor del 99% y 95% de todas las compañías a nivel global (OECD, 2011). En el continente europeo, por ejemplo, alrededor de más de un 93% de la fuerza empresarial, llamadas muy pequeñas empresas, cuentan en su estructura interna con un máximo de 9 colaboradores y, mientras que un 6.5% cuenta con alrededor de 10 y 49 colaboradores como máximo (Moll, 2013). Según Mishra & Mishra (2009), este sector es capaz de generar empleos calificados e ingresos de divisas mediante la exportación de servicios y productos los cuales son elaborados y construidos a distancia para luego ser comercializados.

Según Laporte et al., (2017), en muchas naciones que se encuentran en caminos hacia el desarrollo, el sector del software está dominado por micro o pequeñas empresas, es decir, de 20 empleados o menos, y por empresas jóvenes. En Tailandia, las micro empresas representan alrededor del 90% de la industria del software (Laporte, Muñoz, & Gerançon, 2017). En el Perú, de la totalidad del gremio empresarial que conforman a la industria elaboradora y constructora del software, un 27% tienen el cartel de ser consideradas como pequeñas empresas y, un 63% son consideradas como micro empresas, por lo que, entre ambas, cuentan con más del 90% de la totalidad de empresas de todo este sector, transformándolas en fuerza laboral considerable (PromPerú, 2011).

El desarrollo, construcción y optimización del software es una operación humana intensiva que involucra “la interacción de muchas personas con experiencias, expectativas y conocimientos variados. Es por ello por lo que, la elaboración y construcción del software es una actividad con alta complejidad” (Clarke & O'Connor, 2013). Según Chevers (2017), en función de su complejidad, debe de tenerse un cuidado especial que garantice que los proyectos de desarrollo del software se entreguen a tiempo y dentro del presupuesto y se den cuenta de los beneficios previstos, tendiendo siempre la mirada puesta en la necesidad de entregar productos de software de alta calidad que brinden valor comercial (p.171).

El proceso que conlleva a la construcción de software involucra todas las actividades y las etapas que sigue todo organismo para construir y elaborar un producto software. Se considera que, cualquier proceso de elaboración, construcción y fabricación debe enfocarse en estar siempre actualizado, optimizado y, sobre todo, debe de lograr mantenerse con el fin de alcanzar el cumplimiento de, tanto requerimientos de índole comercial, como de los requerimientos y necesidades de los clientes actuales. Todo ello para evitar problemáticas que desemboquen en costos que vayan en detrimento de las rentabilidades de las compañías.

Acerca de esto, Krasner (2018) menciona que, “el software de mala calidad tiene altísimas implicaciones financieras que afectan a las organizaciones hoy y en el futuro” (pág. 5). Prueba de ello, es que el costo del software de baja calidad en los Estados Unidos durante el año 2018 costó un aproximado de \$2.84 billones, cuyos principales componentes se ven en el siguiente gráfico:

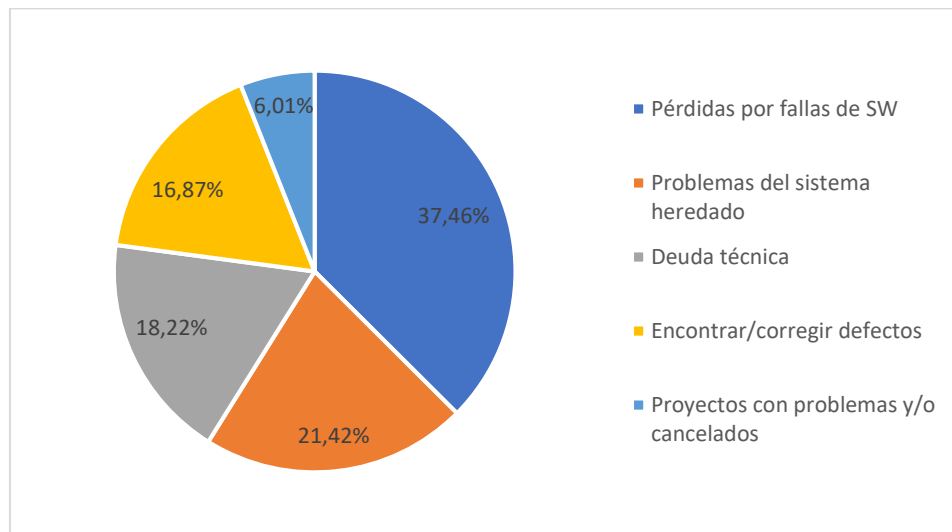


Figura 1, Áreas de costo relacionadas con la deficiente calidad de software en EEUU.

Fuente, Krasner (2018)

Por lo tanto, se debe entender que el incumplimiento y las fallas del software son costosos y puede causar hasta daños de vida humana. Acerca de esto, Quinn (2017) mencionó que, el Ministerio Ejecutivo de Salud y Seguridad del Reino Unido reveló un error de software que puso en duda la veracidad de miles de escaneos médicos durante los últimos 6 años. El sistema de errores, que se utilizaba ampliamente en más de 40 hospitales, amenazaba con invalidar los hallazgos de más de 25.000 radiografías y ecografías, un error que dio lugar a diagnósticos erróneos y tratamientos excesivos o insuficientes. Una vez que se descubrió el error, el fabricante del software emitió una alerta mundial, indicando que el problema técnico podría estar mucho más extendido de lo que se imaginaba. En este caso quedó evidenciado que, el sistema de software costó 40 millones de euros para desarrollarlo y configurarlo, y terminó mostrando la vulnerabilidad a la que estaban expuestos ante su dependencia a la tecnología.

Por tanto, es importante que el software sea intrínsecamente seguro. Y este error del sistema lo identificó claramente.

Asimismo, Frischknecht (2017) reveló que, la empresa de pruebas de software Tricentis analizó 606 fallas de software de una totalidad de 314 empresas para comprender mejor el impacto comercial y financiero de las fallas de software. El informe reveló que estas fallas de software afectaron a 3.6 mil millones de personas y causaron \$ 1.7 billones en pérdidas financieras y un total acumulado de 268 años de tiempo de inactividad. Los errores de software fueron la razón más común detrás de estas fallas. Su informe no asignó fallas a países específicos, pero en términos de PIB relativo, “el 75% de estos totales de fallas son atribuibles a EE.UU., lo que convierte a ese porcentaje en un total de \$ 1.275 billones en cuanto a fallas de software” (Krasner, 2018, pág. 15).

Por otro lado, Burton (2017) mencionó que, Provident Financial, una compañía de préstamos con sede en el Reino Unido que se dirige a clientes de bajos ingresos, admitió que un error de software les impedía cobrar deudas y daría lugar a una pérdida de ingresos de hasta \$ 158,256,000. A la luz de la notificación, disminuyó el precio de las acciones de Provident Financial y estas se desplomaron, perdiendo la friolera de \$ 2.2 mil millones en valor de mercado en un día. El culpable de tal desastre empresarial fue un nuevo sistema de programación de citas destinado a mejorar la eficiencia de los empleados. Los errores en el sistema de programación estaban organizando reuniones en el momento equivocado, lo que hacía que los empleados pierdan reuniones con sus clientes. Esto también puso en peligro a unos 800.000 clientes vulnerables, muchos de los cuales dependen del apoyo que Provident Financial brinda para llegar a fin de mes.

Como diría Frischknecht (2017), “los errores de software nunca duermen”. Por el contrario, los errores de software no perdieron el tiempo dejando su huella durante los últimos años. Según Main (2016), el Ministerio del Interior Británico dijo a miles de estudiantes extranjeros que abandonarían el Reino Unido por un error de software. Después de que una agencia de noticias británica emitiera un

informe de investigación sobre el fraude con respecto a las pruebas de inglés, un requisito previo para visas británicas específicas, el Ministerio del Interior lanzó su propia investigación. Trabajaron con una empresa llamada Educational Testing Services (ETS) para revisar y analizar archivos de voz de exámenes para identificar si las personas estaban haciendo trampa en las pruebas. ETS determinó que 56,419 de los exámenes eran, como mínimo, sospechosos de ser fraudulentos. El Ministerio del Interior había revocado 35,870 visas cuando se descubrió que ETS solo tenía una tasa de precisión de prueba del 80%, lo que significa que alrededor de 7,000 personas inocentes habían sido despojadas de sus visas por error.

Por otra parte, Maki & Sage (2018) revelaron que, un trágico accidente ocurrió, cuando un peatón fue atropellado por uno de los autos autónomos de Uber. La víctima finalmente murió a causa de sus heridas en el hospital. Los vehículos autónomos se comercializan como la alternativa “más segura” a la conducción, ya que utilizan software, tecnología y sensores de vanguardia para prevenir este tipo de accidentes. En este caso, sin embargo, el software no pudo detectar al peatón y el controlador del automóvil no pudo evitar que ocurriera el accidente. Esto muestra que, el software autónomo complejo, como los automóviles autónomos, promete reemplazar de manera efectiva la necesidad de la humanidad de pensar o tomar decisiones en determinadas situaciones y necesita de contar con excelente calidad para que, en este caso específico, pueda salvaguardar la vida de las personas. Un error en el software podría llevarse consigo la vida de las personas.

Relacionado a esto, los distintos modelos y estándares que buscan la gestión y aseguramiento en la calidad del software en el sector constructor de software, se encuentran definidos con el fin de lograr la contribución en la fabricación de productos y/o servicios con altos estándares e índices de calidad, los cuales se encuentren incluidos en el cronograma y los presupuestos fijados, de manera que se optimicen tanto los recursos, como los esfuerzos empleados y que eviten las diversas problemáticas que se generan por esta deficiencia, tal y como se reveló en los párrafos anteriores. Empero, la implantación de praxis

comprobadas implícitas en estos modelos y estándares en ambientes vivenciales de organizaciones de desarrollo de software representan un desafío real, especialmente para micro empresas, las cuales cuentan con hasta 25 personas haciendo que éstas deban realizar un esfuerzo más arduo para salir adelante, y del mismo modo empleando más tiempo y empeño a mejorar sus operaciones y procesos (Muñoz, et al., 2019).

Un sinnúmero de modelos para el proceso constructivo del software se ha postulado con el fin de ser utilizados en organizaciones muy pequeñas, tales como las PYMES, buscando como fin común, el de ser soporte para los procesos de fabricación y producción del software. El proceso de software de personas, denominado comúnmente PSP (por su acrónimo en inglés) y el proceso de software de equipo, denominado comúnmente TSP (por su acrónimo en inglés), son altamente convenientes y recomendados, puesto que logran graficar con criterio a aquellos conceptos que son relevantes, y que son altamente trascendentales para ser utilizados por el modelo de madurez de capacidad integrado (CMMI) correctamente instaurado, el cual ha sido y viene siendo aprovechado especialmente en empresas de mayor jerarquía organizacional, de modo que también pueda aprovecharse en proyectos de menor alcance y envergadura (Chevers, Mills, Duggan, & Moore, 2017). No obstante, la aprobación y acogida para con estos modelos constructivos viene siendo totalmente insuficiente. Para mejorar la adopción, se recomienda el desarrollo y la adaptación de marcos que logren el mejoramiento de aquellos procesos de elaboración y construcción del software según sean los requerimientos y las exigencias de las micro y pequeñas empresas y, sus contextos (Pino, García, Piattini, & Oktaba, 2016). Entre estos modelos existen algunos muy populares tal es el caso de MoProSoft, Competisoft, MPS.BR, OWPL, PRISMS y THEORETICAL FRAMEWORK.

En la praxis, la construcción del software se encuentra plagado ante muchos desafíos, así como también, ante restricciones y dificultades. Aunque existen múltiples enfoques que permiten la organización de los procesos de elaboración y construcción del software y, aunque también, se presentan la existencia de

diversos elementos influyentes en este proceso de fabricación del software, las PYME , por lo general, suelen darle una escasa e insuficiente prioridad a los procesos de elaboración y construcción del software, ya que estas se centran mucho más en los tiempos de entrega del producto o servicio y en la calidad del mismo, mucho más que en la calidad que debe de tener este proceso de construcción (Clarke & O'Connor, 2012). Las PYME de software informan la adopción de marcos de trabajo combinados para los procesos de elaboración y construcción del software, adoptando características de distintos enfoques de la construcción del software prescritos anteriormente, lo cual les permite lograr la satisfacción de sus requerimientos, a pesar de las diversas dificultades que esta puede tener en diferentes aspectos. Según Coleman & O'Connor (2008), las micros y muy pequeñas empresas carecen de modelos de procesos que se han desplegado para la fabricación y elaboración del software, los cuales les sirvan de guía, prefiriendo en cambio dejar caer elementos de su modelo elegido o, desarrollar algo patentado que mejor se adapte a sus requerimientos específicos. Asimismo, todos los manuales y procedimientos de trabajo de la ingeniería de software con los que cuentan, se eligen de forma oportunista, algunas veces adaptados y configurados de modo que proporcionen un valor, sin embargo, bajo las diversas limitaciones que su contexto le impone en tal momento (Vijayarathy & Butler, 2016). Efectivamente, las compañías actuales vienen mostrando señales de adopción, para con esta diversidad de métodos y metodologías, en los distintos proyectos y eligiendo el seguimiento a un enfoque híbrido en los procesos de elaboración y construcción del software.

Se encuentra evidenciado por varios estudios nacionales que, un gran porcentaje de las pequeñas entidades productoras de software, no se encuentran siguiendo, ni mucho menos practicando los estándares existentes ya que perciben que están orientados para empresas de mayor relevancia y envergadura y, la revisión bibliográfica ha dejado demostrado que, la percepción negativa de las micro y pequeñas empresas sobre las normas y estándares del modelo del proceso de elaboración y construcción de software son principalmente impulsadas por puntos de vista negativos de costo, documentación y burocracia (Laporte, Alexandre, & O'Connor, 2008).

Acerca de lo mencionado anteriormente, Amable & Millones (2019) encontraron en su estudio para la Universidad de Lima que, un 53.3% de un total de 329 micro y pequeñas organizaciones dedicadas a la construcción de software en la ciudad de Lima fundamentaban sus procesos constructivos en modelos debidamente certificados tales como las ISO/IEC 12207 e ISO/IEC 29110 o los modelos Competisoft, Moprosoft y CMMI. Esto revelaba que, el otro porcentaje de 46.7% fundamentaba sus operaciones de construcción de software en prácticas propias ajustadas a sus requerimientos pero que no se basaban en dichos estándares certificados, tal y como se muestra en la siguiente figura:

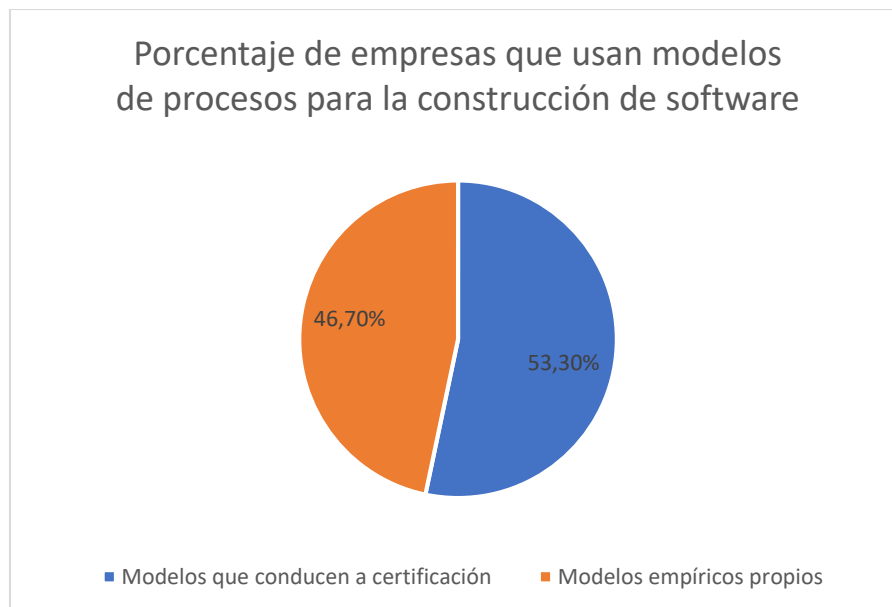


Figura 2, Porcentaje de empresas que usan modelos de procesos.

Fuente, Amable & Millones (2019)

Por este motivo, es importante garantizar la idoneidad con la que cuentan los productos de las PYME mejorando la calidad de los procesos de elaboración, construcción y optimización del software usados para desarrollarlos. No obstante, las PYME sufren la falta de implantación de procesos formales para la construcción de software. Por ello, esta investigación se ha argumentado con el fin de desarrollar un modelo de procesos hecho estratégica y exclusivamente para la construcción del software en las microempresas desarrolladoras de software en el Perú y en pequeñas organizaciones de menos de 25 colaboradores.

1.2. Antecedentes de Estudio.

Suteeca (2020), realizó la investigación, *A Software Process Gap Analysis Methodology for Very Small Entity*, en la Chiang Mai University, en Chiang Mai, Tailandia. Hoy en día, la demanda de TI y software viene en aumento, estableciendo muchas más organizaciones desarrolladoras de software, las cuales padecen muchos problemas en dichas labores. Por esta razón, se evaluaron y analizaron los procesos de referencia de pequeñas organizaciones por el estándar ISO/IEC 29110, para posteriormente diseñar una metodología para mostrar la brecha entre el proceso actual y el estándar. El resultado de la evaluación condujo a 10 pequeñas empresas tailandesas de software a priorizar el riesgo cuando implementan el estándar para la organización, demostrando también que, la brecha más importante en el procedimiento y planificación de software es la identificación y monitoreo de los peligros, mientras que, la fortaleza de dichas compañías es la entrega de productos y, finalmente, el diseño detallado de software es la brecha más baja en el proceso de implementación de software. Se concluyó que, a pesar de que los proyectos de software son difíciles y complicados al momento de gestionar, la calidad de los productos no debe dejarse de lado, considerando que para el cliente es lo más importante, por lo que los estándares tales como CMMI, ISO/IEC 12207 y la ISO/IEC 29110 son soporte fundamental para el mejoramiento de sus procesos constructivos de software.

Bayona et al. (2019), realizaron la investigación, *Software Process Improvement: Requirements Management, Verification and Validation*, en la Universidad Autónoma del Perú, en Lima, Perú. Actualmente las organizaciones requieren de soluciones informáticas que les permitan estar a la vanguardia del ámbito empresarial donde se desarrollan y que les permita brindar productos/servicios de calidad cumpliendo fielmente con las exigencias de su clientela, por lo que la usanza de estándares internacionales para la mejora de sus procesos productivos de software es relevante, verbigracia, CMMI. Por esta razón, se aplicó CMMI en un área de proyectos de una escuela privada de la ciudad de Lima, con el propósito de mejorar sus procesos productivos, minimizando errores y defectos mediante una buena praxis en lo que respecta

a pruebas de software. Los resultados alcanzados mostraron que, dicha área se encontraba en con 63.44% en cuanto a Gestión de Requisitos, 48.93% en cuanto a Verificación, 52.20% en cuanto a validación, demostrando así la necesidad de lograr el mejoramiento de dichos procesos mediante la implementación de CMMI. Se concluyó que, con la implementación de dicho modelo, se mejoraron las tres (03) áreas de procesos mencionadas anteriormente, mejorando la calidad de los productos/servicios, aminorando las problemáticas acaecidas en los procesos constructivos y reduciendo las tasas de defectos en las pruebas realizadas posteriormente.

López et al. (2019), realizaron la investigación, *Bringing a Micro Enterprise towards the ISO 29110*, en la Universidad Nacional Agraria de la Selva, en Tingo María, Perú. A Nivel mundial se ha visto un crecimiento en lo que respecta a la industria de la construcción de software, en la cual, las pequeñas organizaciones y/o pymes han tomado un rol protagónico, por lo que requieren de estándares para la mejora en sus productos/servicios presentando una excelente calidad y obteniendo una mayor competitividad, por lo que la ISO/IEC 29110 aparece como un socio idóneo para tal fin, empero, aún hay una brecha acerca de qué herramientas utilizar para dicho proceso de alinearse con dicha norma. Por esta razón, se buscó elegir un conjunto de herramientas que permitan facilitar la adopción de dicho estándar en tales compañías menores, haciendo usanza de un caso de estudio, una empresa de 5 programadores quienes trabajaron en un proyecto de software para restaurantes por un lapso de 4 meses. Los resultados obtenidos mostraron que, con la usanza de un grupo de herramientas conformados por VSTS, Google Drive, Gmail, IDE Eclipse, UML2 Plugin y Jenkins se logró cubrir el 90% de los proyectos de implementación de software, mientras que el 80% de estas herramientas contaron con una valoración muy satisfactoria, esto permitió que las personas involucradas en dicha entidad no necesitarán del aprendizaje de herramientas nuevas. Se concluyó que, con la implementación y usanza de estas herramientas no fue necesario alterar de modo drástico los procesos con los que contaba dicho caso de estudio anteriormente.

Mahmoud et al. (2019), realizaron la investigación, *Harmonized Software Quality Improvement Models for Sudanese SME Based on CMMI*, en la Nile Valley University, en Atbara, Sudán. Las compañías desarrolladoras de software implementaron más de un modelo de procesos para lograr un alto valor en sus productos y servicios, empero, se necesitan más recursos para poder respaldar dicho trabajo por lo que, esto no está disponible en las pequeñas organizaciones especialmente en las sudanesas. Por esta razón, desarrollaron un modelo de procesos para el desarrollo de software aplicable a mypes sudanesas basado en CMMI-DEV y la ISO/IEC 25010 con el propósito de alentarla a mejorar sus procesos. Los resultados obtenidos mostraron que, la mayoría de las empresas de software sudanesas no seguían modelos claros de SPI o estándares de calidad de software, y el éxito de estas instituciones depende del heroísmo de sus equipos. Se concluyó que, las instituciones de desarrollo de software sudanesas que necesitan mejorar la calidad de sus productos de software deben concentrarse en analizar tareas para conocer el atributo de calidad requerido. Por lo tanto, esto se verá como un movimiento de apertura para integrar tanto la calidad del software como el proceso de desarrollo de software. Entonces será posible estudiar modelos de software, en lo que respecta al proceso organizativo y los productos de calidad de software necesarios para elegir las prácticas más adecuadas.

Muñoz et al. (2019), realizaron la investigación, *A Comparative Analysis of the Implementation of the Software Basic Profile of ISO/IEC 29110 in Thirteen Teams That Used Predictive Versus Adaptive Life Cycles*, en la Conferencia Europea sobre Mejora de Procesos del Software, en la ciudad británica de Edimburgo, Escocia. El incremento en la demanda de software ha creado oportunidades para las empresas y equipos desarrolladores de software, con especial énfasis en aquellas organizaciones muy pequeñas, puesto que, una gran cantidad de ellas producen software para empresas medianas y grandes como parte de una cadena de producción que proporciona servicios y productos de manera que se logre la satisfacción de la demanda con la que cuenta actualmente el mercado global. Ante ello, existe la imperiosa necesidad de lograr el mejoramiento y el perfeccionamiento de sus procesos de elaboración

y construcción del software, lo que permita mantenerse en el negocio a través del desarrollo de servicios y productos de software de calidad con recursos limitados. El problema yace en la carente y escasa percepción de conocimiento y experiencia práctica con respecto a implementación de los modelos para alcanzar el mejoramiento y optimización de los procesos constructivos de software actuales y los estándares de ingeniería de software ya existentes. Por esta razón, se realizó un análisis comparativo de trece empresas desarrolladoras de software cuyos equipos de trabajo desarrollan productos haciendo uso de metodologías predictivas y adaptativas durante el proceso de implementar el perfil básico de software de la ISO/IEC29110. Entre los resultados obtenidos destacan que, la mayor parte de los equipos que se encargan del proceso constructivo del software cuentan con un porcentaje medio y alto en cuanto a la carencia en el cumplimiento de prácticas, en lo que respecta con el perfil básico del estándar internacional ISO/IEC29110. Además, la mayoría de los equipos tienen una mejor cobertura de los procesos de Gestión de Proyecto decantándose un 77% en la utilización de prácticas ágiles. Esta investigación logró demostrar que los modelos y estándares en el proceso constructivo del software, son llevados a cabo el fin de lograr aportar conocimiento en la fabricación de servicios y/o productos de calidad, los cuales sean entregados en los plazos establecidos previamente en el cronograma inicial y se encuentren también dentro del presupuesto fijado, logrando de esta manera la optimización de recursos y esfuerzos. No obstante, se debe realizar e implementar estas prácticas comprobadas de estos modelos y estándares en organizaciones de desarrollo de software, deben trabajar más duro para sobrevivir, y dedicarle mucho esfuerzo y tiempo al mejoramiento de sus procesos y operaciones, es un verdadero desafío.

Chevers (2017), realizó la investigación, *Software process improvement: awareness, use, and benefits in Canadian software development firms*, en la Revista de Administración de Empresas, de la Escuela de Administración de Empresas de Institución Getúlio Vargas, en Sao Paulo, Brasil. Desde hace décadas las empresas desarrolladoras de software se han venido preocupando por entregar productos y/o servicios de calidad. Ante ello, el mejoramiento de

procesos de software busca consolidar una iniciativa que propicie la entrega de productos con altos estándares de calidad. A pesar de estos denodados esfuerzos, las tasas de uso, conocimientos y beneficios otorgados por estas inventivas para lograr el mejoramiento de los procesos de la elaboración y construcción del software en las empresas canadienses vienen siendo escasas. Por esta razón, este estudio empleó un enfoque de encuesta en línea para recopilar datos de profesionales de Ingeniería de Sistemas que permitieran determinar si la baja iniciativa en estos procesos de mejora del software en las compañías de origen canadienses repercute en los niveles de calidad del producto y calidad del proceso y, posteriormente en el beneficio de la organización. Los resultados adquiridos indicaron que, el 59% de los personajes encuestados conocía los modelos para el mejoramiento y optimización de los procesos de elaboración y construcción del software y el 41% no. Entre las razones de los encuestados para no estar al tanto, y desconocer de manera rotunda, acerca de los modelos para el mejoramiento y optimización de los procesos de elaboración y construcción de software figuran el no estar al tanto del concepto, el tamaño organizacional de las compañías es considerado relativamente pequeño y no necesitar dichos conceptos en el desarrollo de software. Asimismo, solo el 43% de los encuestados usó alguna forma de programas que busquen el mejoramiento y optimización de procesos en el desarrollo de software, y el 57% no utilizó ninguno de estos programas. Los resultados de este estudio resaltaron la relevancia del mejoramiento de los procesos de elaboración y construcción del software como un medio que permita obtener beneficios operativos específicos, de modo que, aumenten la capacidad y la madurez de dichos procesos para producir productos de software de mayor calidad y ser competitivos en el mercado global.

Chevers et al. (2017), realizaron la investigación, *Toward a simplified software process improvement framework for small software development organizations*, en la Revista de Gestión Global de Tecnología de la Información, en Philadelphia, Estados Unidos. Durante varios decenios, los equipos fabricantes y constructores de software se han encontrado en la lucha por lograr el ofrecimiento de sistemas que obedezcan con el propósito de contar

altos estándares en calidad. Para abordar los problemas relacionados con el software de baja calidad, muchos utilizan modelos para la perfección y optimización de los procesos de elaboración y construcción de software (SPI) como la integración del modelo de madurez de capacidades integrado (CMMI) con el fin de alcanzar orientar el proceso de elaboración y construcción del software. No obstante, lograr la implementación e impulso de estos ingenieros e inventivas en muy pequeñas empresas del rubro de la construcción y fabricación de software tiende a convertirse en un desafío muy particular, toda vez que existe un alto nivel en cuanto a nivel de complejidad y en cuanto al coste involucrados en dichos procesos. Por este motivo, este tratado propuso un marco de mejoramiento sintetizado de los procesos de elaboración y construcción del software, basándose en el modelo CMMI, que las organizaciones pequeñas pueden usar para favorecer lograr la mejora de la construcción del software. Llevado a cabo en los países caribeños de Trinidad y Tobago, Jamaica, Guyana y Barbados, esta investigación utilizó el método de grupo focal echando mano de ingenieros de sistemas con vasta experiencia, todo ello con la finalidad de lograr el objetivo de alcanzar la aprobación indiscutible por parte de ellos para con el contenido y contando con un modelo propuesto, este nos lleva al perfeccionamiento de los procesos constructivos de software en estas naciones caribeñas. Los resultados obtenidos mostraron que, este nuevo modelo resultante, denominado modelo SPI para organizaciones pequeñas (SPIM-S) se consideraba menos engorroso y complejo de implementar, en comparación con su contraparte, el CMMI, al mostrar una reducción del 45% en el número de prácticas en comparación con el CMMI completo, para los niveles 1 a 3. El modelo planteado, correspondería a uno que busca el menor perjuicio al momento de la implementación, de manera que ofrezca a distintos profesionales un soporte alternativo en el sondeo de lograr un software con altos estándares y niveles de calidad.

Khan et al. (2017), realizaron la investigación, *SPIIMM: toward a model for software process improvement implementation and management in global software development*, en la City University of Hong Kong, en Kowloon Tong, Hong Kong. Las organizaciones de desarrollo de software están globalizando

sus actividades adoptando el fenómeno del desarrollo global de software (GSD), principalmente debido al importante retorno de la inversión que ofrece, empero, varios desafíos están asociados con la mejora de procesos de software (SPI). Empero, se han creado modelos y estándares de mejora de procesos en el contexto GSD, los cuales no han sido abordados y desarrollados ampliamente y, por lo tanto, los esfuerzos de mejora de procesos han tenido un éxito limitado. Por esta razón, se desarrolló un modelo de gestión e implementación de mejora de procesos de software al que denominaron SPIIMM, el cual sirviera como soporte para ayudar a tres compañías a evaluar y mejorar sus actividades de SPI y el cual estuvo fundamentado en tres (03) modelos: CMMI, IMM y SOVRM. Los resultados obtenidos mostraron que, SPIIMM es una herramienta adecuada para que las organizaciones evalúen sus niveles de madurez relacionados con SPI y guíen a una organización para mejorar sus actividades de SPI a niveles avanzados. Asimismo, mediante una revisión de estudios se identificaron, 21 factores de éxito (impacto positivo) y 22 barreras (impacto negativo). Se concluyó que, SPIIMM es una herramienta adecuada para que las organizaciones evalúen sus niveles de madurez relacionados con SPI y guíen a una organización para mejorar sus actividades de SPI a niveles avanzados.

Laporte et al. (2017), realizaron la investigación, *The education of students about ISO/IEC 29110 software engineering standards and their implementations in very small entities*, en la IEEE Conferencia Internacional de Tecnología Humanitaria de la ciudad de Toronto, en Ontario, Canadá. Alrededor del mundo, existe un gran porcentaje, por no decir el 95%, de compañías fabricadoras y constructoras del software que son organizaciones empresariales cuyo capital humano no sobrepasa los 25 trabajadores. Dado que la mayoría de los estándares internacionales enfocados en la ingeniería de software publicados, tales como la ISO/IEC 12207, habían sido desarrollados por y para grandes organizaciones, la mayoría de microempresas constructoras de software no tienen la experiencia para adaptar esos estándares de ingeniería de software de modo que puedan satisfacer sus necesidades, además de no disponer de recursos en términos de experiencia y dinero. Por esta razón, esta investigación mostró una orientación de la ISO/IEC 29110 con respecto a la gestión e

ingeniería con el propósito de ayudar a las micro empresas, presentando ejemplos de su implementación en países como Tailandia, México, Haití, Perú, Túnez y Colombia, puesto que un gran porcentaje de países en vías de desarrollo aún no conocen esta norma mundial. Los resultados obtenidos demostraron que, distintas naciones lograron adoptar como un estándar de carácter nacional a la ISO/IEC29110. Asimismo, también se pudo determinar que, algunos países están ayudando a sus micro empresas constructoras de software en la adopción, en las actividades de implementación y certificación con programas gubernamentales, tal es el caso del Perú que cuenta con una Oficina Nacional de Gobierno Electrónico e Informática (ONGEI por su acrónimo). Los esquemas para certificación y evaluación independientes de bajo costo permiten a las micro empresas de todo el mundo demostrar el reconocimiento de sus competencias a clientes y socios locales e internacionales.

Sánchez et al. (2017), realizaron la investigación, *The Route to software process improvement in small and medium-sized enterprises*, en el libro *Managing Software Process Evolution: Traditional, Agile and Beyond - How to Handle Process Change*, Suiza. Las micro y pequeñas empresas se muestran como un grupo de mayor dominancia en el total de compañías que pertenecen a la industria del desarrollo de software en su totalidad, por lo que constituyen un porcentaje superior al 95% y en algunos casos puntuales, llegando a las cifras exorbitantes del 99% del sector empresarial en algunos países. Ante ello, es necesaria la implantación de estructuras y controles que permitan administrar de una manera adecuada y eficaz sus actividades de desarrollo de elaboración y construcción del software. Esto, verdaderamente, establece un gran desafío. En este sentido, una forma común de lograr el desarrollo de un software con elevados y, sobretodo, excelentes estándares de calidad, es mediante la introducción de un modelo de procesos de elaboración y construcción del software adecuado. Por esta razón, se desplegó un análisis en el cual se caracterizó a las pymes constructoras y fabricantes de software, así como también, una descripción general de varios modelos, teniendo en consideración los procesos de elaboración y construcción de software y modelos de

perfeccionamiento y optimización de los procesos de software (SPI), que están destinados a ayudar a las PYME a mejorar la forma en que desarrollan software. Los resultados obtenidos mostraron que, existían estándares de calidad tales como CMMI, ISO 9001, ISO/IEC12207, ISO/IEC15504 e ISO/IEC 29110 y, además también, marcos de referencia tales como ITMark, Tutelkan, MoProSoft, MPS.BR y COMPETISOFT. Teniendo en cuenta la numerosa diversidad de configuraciones de construcción del software, la ruta a la mejora y optimización del desarrollo de software en las pymes depende de la cantidad de recursos, esfuerzo y objetivos de cada uno. Por una parte, el bajo costo, la ligereza del proceso y la facilidad de uso a través de herramientas de carácter automatizado son rasgos distintivos relevantes. Por otra parte, el patrocinio y el respaldo por parte de organizaciones de carácter mundial como la ISO y, de gobiernos de nivel local y nacional, son elementos sumamente relevantes.

Díaz et al. (2016), realizaron la investigación, *The ISO/IEC 29110 Implementation on two Very Small Software Development Companies in Lima. Lessons Learned*, en la Pontificia Universidad Católica del Perú, en Lima, Perú. Actualmente, y a nivel mundial, las pymes desarrolladoras de software son el núcleo fundamental de la industria del software sosteniendo un nivel de participación de más del 90%, enfocándose en la mejora de sus procesos constructivos, por lo que enfocan sus esfuerzos en la usanza de modelos tales como ProCal-ProSer, CompetiSoft, RELAIS, CMMI, ISO/IEC 12207, etcétera, empero dichos modelos han sido difíciles en sus procesos adaptativos a las coyunturas de cada compañía. Por esta razón, se recogieron experiencias con respecto a la mejora de procesos y se pudo obtener un aprendizaje de estas lecciones aplicando el perfil básico y organizativo de la ISO/IEC 29110 en dos (02) pequeñas organizaciones limeñas desarrolladoras de software denominadas Lim.Epsilon y Lim.Nu. Los resultados obtenidos mostraron que, Lim.Epsilon logra el nivel de capacidad de proceso de 1 en su proceso de PPM y mantiene la calificación en el proceso PM. Por otro lado, Lim.Nu no logra llegar al nivel de capacidad de proceso en 1 en ningún proceso, pero se encuentra bastante cerca pues sus niveles de adhesión están cercanos al 85% tanto en PM, SI y PPM. Se concluyó que, en Lim.Epsilon y Lim.Nu, la definición y

ejecución de los procesos que priorizaron para el ciclo de mejora permitieron, en ambos casos, mejorar los procesos RM y PSM. Entre las tareas que contribuyeron a la mejora se tuvieron: la identificación de recursos según necesidades, la definición de un repositorio organizacional, la planificación y realización de entrenamientos, gestión de proveedores y la identificación y definición de algunos procesos.

Larrucea, O'Connor, Colomo & Laporte (2016), realizaron la investigación, *Software Process Improvement in Very Small Organizations*, en la Revista IEEE Software. Al pensar en compañías desarrolladoras de software, por lo general se piensa en grandes empresas, sin embargo, la realidad es otra. Las microempresas, las cuales cuentan con hasta 25 trabajadores, cuentan con un gran flujo de productos y servicios, por lo que, las convierte en la fuerza dominante de la industria mundial del software y son cruciales para su competitividad e innovación. Por ello, es necesario que cuenten con mejora de procesos de software, sin embargo, éstas carecen de los mismos ya que, existen modelos tales como. el caso del modelo CMMI y la ISO/IEC 15504, los cuales han sido desarrollados para grandes compañías. Además, que la percepción de que estos esfuerzos fueron desarrollados por y para organizaciones más grandes, son costosas, requieren mucha documentación y burocracia, y no establecen claramente los procesos de software. Por esta razón, este estudio construyó una fábrica de experiencias que permitió ayudar a las microempresas a comenzar iniciativas para el mejoramiento del desarrollo de los procesos de elaboración y construcción del software. Se analizaron 74 pymes y se determinó que, cumplen con los requisitos del estándar internacional ISO/IEC 29110. Esto indicó que, un porcentaje muy elevado de microempresas constructoras de software ya contaban con procesos que satisfacían algunos requisitos básicos y no deberían enfrentar muchas dificultades para cumplir plenamente. Esta investigación logro establecer que, los componentes del software, el diseño minuciosamente al detalle y la arquitectura del software, la verificación y validación, la implantación del software, los casos de estudio para las pruebas y los procedimientos de las pruebas, son aquellas áreas que exhiben muchas mayores contingencias e inconvenientes. Por esta razón, las

pymes deben primero identificar un objetivo para sus iniciativas de mejoramiento de sus procesos de elaboración y construcción del software y luego pueden elegir enfoques tradicionales o más innovadores, dependiendo de factores como sus necesidades y las tendencias actuales de la industria.

Suteeca & Ramingwong (2016) realizaron la investigación, *A framework to apply ISO/IEC 29110 on SCRUM*, en la Chiang Mai University, en Chiang Mai, Tailandia. Durante los últimos años, las pequeñas empresas de software han mostrado interés en utilizar estándares de desarrollo de software para mejorar sus procesos y competitividad. Sin embargo, esto puede ser un inconveniente para las empresas que no tienen experiencia en la mejora de procesos estándar o de software (SPI)., por lo que la ISO/IEC29110 es un estándar de desarrollo de software ligero que incluye la gestión de proyectos y el proceso de implementación de software. Por esta razón, implementaron la ISO/IEC 29110 versión 2011, perfil básico, en aplicación con la metodología SCRUM. Los resultados obtenidos mostraron los marcos prácticos de las actividades de cada una de ella en la implementación de software. Se concluyó que, la ISO/IEC 29110 cubre todos los aspectos de la implementación de software en Scrum, excepto una sola actividad, define el backlog del producto. Por otro lado, el desarrollo de Scrum satisface todas las facetas de la implementación de software de ISO / IEC29110. Esto podría alentar mucho a los practicantes de Scrum a que su desarrollo eventualmente pueda alinearse con este estándar.

Espejo, Bayona & Pastor (2016), realizaron su investigación, *Quality assurance in the software development process using CMMI, TSP and PSP*, en la Reveista Ibérica do Sitemas e Tecnologias da Informação - RISTI, en la ciudad de Rio Tinto, Portugal. Actualmente las compañías constructoras de software vienen convirtiéndose en socios de carácter estratégico de sus usuarios y clientes mediante la creación de valor, haciendo uso valioso de ofertas de servicios y de productos que logran ser el soporte de la diversidad de los procesos de negocio. A pesar de ello, fluyen innumerables inconvenientes en los procesos de elaboración y construcción del software a nivel de calidad en los procesos mismos. Ante ello, es necesario la implementación y la usanza de

estándares de nivel internacional, los mismos que permiten asegurar las entregas con productos o servicios de software con elevados estándares en cuanto a calidad. Por esta razón, en esta investigación se describió la implementación de un modelo que permitió el fortalecimiento de la calidad para los procesos de elaboración y construcción del software, el mismo que estuvo integrado por Personal Software Process (PSP), Team Software Process (TSP) y el modelo Capability Maturity Model Integration (CMMI). Dicho modelo fue implementado a la interna de una compañía peruana constructora y fabricante de software, la misma que, ya había logrado alcanzar la certificación anteriormente en CMMI, el mismo que le permitió el mejoramiento y el fortalecimiento de la calidad del proceso de elaboración y construcción del software. Los resultados obtenidos mostraron que, existían mejoras de gran significancia, tales como la disminución en la cantidad de defectos y en los costos de calidad de los proyectos. Igualmente ocurría con la satisfacción de los clientes, la cual se veía incrementada junto a la rentabilidad de los proyectos. Se concluye que esta investigación fue relevante puesto que, se estudió el impacto en los proyectos de elaboración y construcción de software. Asimismo, fue sumamente relevante ya que, mostró las cuatro dimensiones de la variable dependiente denominada calidad en el proceso de construcción del software, los cuales son: el coste de la calidad por proyecto, los defectos acaecidos por proyecto, la rentabilidad del proyecto mismo y la satisfacción percibida por el cliente, toda vez que dicha variable y sus dimensiones servirán de aporte en la operacionalización de las variables.

Chevers, Mills, Duggan, & Moore (2016), realizaron su investigación, *An Evaluation of Software Development Practices among Small Firms in Developing Countries: A Test of a Simplified Software Process Improvement Model*, en la Revista de Gestión de la Información Global – JGIM, en Pensilvania, Estados Unidos. Cuando las compañías fabricantes y constructoras del software buscan competitividad, deben también dar garantía de su calidad en el servicio o producto que brindan. Partiendo de esto, estas compañías buscan la adopción de modelos que les permitan mejorar su proceso constructivo y desarrollador del software, tal como el caso del CMMI-DEV. No

obstante, estos modelos de procesos de mejoramiento suelen ser de alto coste y totalmente engorrosos, sobre todo para aquellas empresas fabricantes del software que pertenecen a países que se encuentran en economías en vías de desarrollo y que cuentan con una cantidad limitada de recursos. Por esta razón, esta investigación asumió como finalidad proponer un modelo de mejoramiento y optimización de los procesos de elaboración y construcción de software simplificado enfocado en pequeñas empresas, denominado (SPM-S), el cual estuvo compuesto por 10 prácticas claves de desarrollo de software con menos prácticas, de modo que, el modelo propuesto debía ser más accesible y menos costoso de implementar. Los resultados obtenidos mostraron que, si bien diez (10) áreas de práctica se consideraban una prioridad (por lo que se incluyeron en el modelo simplificado de mejoramiento de los procesos de elaboración y construcción del software para micro y pequeñas empresas, SPIM-S), para las empresas que participaron en este estudio, solo tres prácticas tenían un impacto de manera significativa en la calidad del software: enfoque del proceso dentro de la organización, descripción y detalle del proceso de la organización y, verificación y validación. Empero, esta situación no refleja y demuestra que las otras prácticas no cuenten con la relevancia debida, sino que su explicación no ha sido significativa en cuanto a términos de calidad de software, por lo que no han sido consideradas en el contenido de esta investigación.

1.3. Teorías relacionadas al tema.

1.3.1. Sistema de gestión de la calidad de software

Un sistema de gestión de la calidad (SGC), comúnmente nombrado como sistema de calidad, es una inventiva que es usada por compañías e instituciones, con el fin salvaguardar la calidad esperada de los productos que construyen o fabrican. Según Mall (2014), algunos de los problemas importantes asociados con un sistema de calidad son:

a. Estructura gerencial y responsabilidades individuales

Un SGC es obligación de toda compañía e institución en toda su extensión. No obstante, cada compañía e institución cuenta con un área encargada del aseguramiento de la calidad, la cual se encuentra

separada, de manera que realice diversas acciones del sistema para la gestión del proceso adecuado que buscar asegurar, con una práctica apropiada, la gestión de calidad. El SGC de una organización debe contar con la plena aprobación de la alta gerencia y dirección. Si no se contara con el soporte brindado por sistema de gestión que permita asegurar y certificar la calidad en un máximo y óptimo nivel a la interna de una compañía o institución, verdaderamente que, serían escasos aquellos colaboradores que tuvieran en consideración este tema tan importante como lo es el de contar con un sistema de calidad a todo nivel (Mall, 2014).

b. Actividades del sistema de calidad

Según Mall (2014), las actividades que involucra un sistema para la búsqueda de la gestión del aseguramiento de la calidad incluyen las siguientes:

- i. Auditoría de proyectos para verificar si se siguen los procesos.
- ii. Recopile métricas de procesos y productos y analícelos para verificar si se cumplen los objetivos de calidad.
- iii. Revisión del sistema de calidad para hacerlo más efectivo.
- iv. Desarrollamiento de pautas, procedimiento y normas.
- v. Producir informes con la finalidad de ser expuesto a la alta dirección de la compañía, los cuales deben de resumir la eficacia del sistema para la labor de gestionar el aseguramiento de la calidad al interior de una compañía.

El SGC que existe en toda organización debe de ser documentado en todo su esplendor. En este caso, si se contara con un SGC, el carecer de documentación, no permitiría poder llevar a cabo el desarrollamiento y la aplicación de cualquier procedimiento o control de calidad, lo que conllevaría a enormes varianzas en la calidad entregada en cada uno de sus servicios o productos. Asimismo, no documentar el sistema anteriormente mencionado, conlleva al envío de cualquier mensaje erróneo a todos los

colaboradores en que, la compañía muestra una condición poco inclinada hacia el aseguramiento de la calidad en cualquiera de sus procesos y sobretodo, en la entrega de sus productos y/o servicios. “Los diversos estándares de carácter internacional, tales como el caso de la ISO 9000:2015, suministran y orientan acerca del proceso de organización y clasificación de un sistema para la búsqueda de la calidad a todo nivel” (Mall, 2014, pág. 128).

1.3.1.1. Evolución de los sistemas de calidad

Los sistemas involucrados en la gestión de la calidad vienen evolucionando desde hace muchísimo tiempo, y lo vienen realizando una manera fugaz y acelerada, sobre todo durante los últimos setenta años. Precedentemente a la Segunda Guerra Mundial, el procedimiento más tradicional que se venía desarrollando para la producción con calidad era llevar a cabo la inspección de los productos terminados con el fin de llevar a cabo la eliminación de aquellos productos que tuviesen algún tipo defecto. Por ejemplo, una empresa que fabrica tuercas y tornillos inspeccionaría sus productos terminados y rechazaría aquellas tuercas y tornillos que están fuera de cierto rango de tolerancia especificado.

Desde aquel tiempo hasta el día de hoy, los sistemas de calidad en las compañías e instituciones han logrado experimentar cuatro (04) fases evolutivas con el pasar de los años, tal y como puede observarse en la **Figura 3**. El método de inspección del servicio y/o producto, fue el paso inicial que permitió comenzar con aquellos principios del control de la calidad (QC).

Este concepto de QC no solo se centra en la detección de productos con algún tipo de defecto para luego proceder a su eliminación, sino que también logra el establecimiento de las causas que anteceden y producen estos defectos, con el fin de lograr la reducción de tasas con porcentaje de rechazo de los productos.

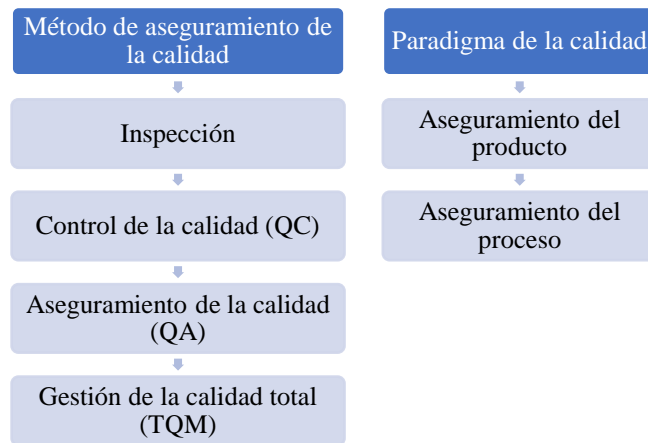


Figura 3, Evolución del método de aseguramiento de la calidad y cambio en el paradigma de la calidad.

Fuente, Mall (2014)

Por lo tanto, el control de calidad tiene como objetivo corregir las causas de los errores y no solo rechazar los productos defectuosos. La siguiente fase en la evolución del SGC ha sido la construcción del principio de garantía de calidad (QA). Relacionado con ello, Mall (2014), señala lo siguiente:

La proposición principal de garantía moderna de calidad menciona que, si es que el proceso para el desarrollo de un producto o servicio en una compañía o institución, es bueno y, además, es llevado a cabo con alto nivel de rigurosidad y precisión, pues en este caso, estos productos o servicios, definitivamente van a lograr ser de calidad óptima y que cumplirán con los requerimientos del usuario final (p.623)

La última fase en la evolución del SGC, es el modelo moderno para el aseguramiento de la calidad, en sobre el cual se encuentra incluida las recomendaciones para el reconocimiento, la definición, el análisis la mejorara del proceso productivo. La gestión de calidad total (TQM) considera que cualquier proceso que una compañía o institución sigue debe ser mejorado paulatinamente y de manera continua, realizando evaluaciones a dicho proceso. La gestión de

calidad total, siempre se encuentra un paso más adelante del aseguramiento de la calidad, por lo que este concepto busca en todo momento el mejoramiento continuo de todos los procesos. La gestión de calidad total se encuentra un paso más adelante que solamente lograr la documentación de los procesos, sino que busca rediseñarlos y reinventarlos con el fin de lograr la optimización de los mismos. Un término relacionado con TQM es reingeniería de procesos comerciales (BPR), cuyo objetivo es rediseñar la manera en cómo se llevan a cabo los negocios en una organización, mientras que el enfoque de este texto es la reingeniería del proceso de elaboración y construcción del software.

De la discusión anterior, se puede decir que en las últimas seis décadas más o menos, el paradigma de la calidad ha pasado de la garantía del producto a la garantía del proceso, tal y como puede verificarse en la **Figura 3**.

1.3.1.2. Métricas de producto versus métricas de proceso

Todos los sistemas de calidad modernos ponen énfasis en la recolección de ciertas métricas de productos y procesos durante el desarrollo del producto. Primeramente, entiéndase aquellas incompatibilidades primarias que se dan entre las métricas de los procesos y las métricas de los productos. Acerca de esto Mall (2014), menciona que:

Las métricas de productos ayudan a medir las características de un producto que se está desarrollando, mientras que las métricas de procesos ayudan a medir el rendimiento de un proceso (p.624)

Ejemplos de métricas del producto son, las líneas de código (LOC por su acrónimo en inglés) y, punto de función para medir el tamaño, PM (persona-mes) para medir el esfuerzo requerido para desarrollarlo, meses para medir el tiempo requerido para desarrollar

el producto, la complejidad temporal de algoritmos, etc. Ejemplos de métricas de proceso son, la cantidad promedio de fallas encontradas durante el proceso de pruebas, el tiempo promedio de la corrección de los defectos, la tasa productividad, la cantidad promedio de los defectos encontrados por hora en el procedimiento de inspección, la efectividad de la revisión, la cantidad de los defectos latentes por línea de código en el producto o servicio elaborado, etc.

1.3.2. Microempresas constructoras de software

1.3.2.1. Microempresas de software

Teniendo en cuenta que a nivel internacional se categoriza a los servicios informáticos y para software, hay pocos datos oficiales internacionales disponibles fuera de Europa y América del Norte. Según Sánchez et al. (2017), en Europa, la Oficina Europea de Estadística, conocida como EUROSTAT por su acrónimo en inglés, se vale de la Clasificación Industrial General de las Actividades de tipo Económico que rige a la interna de las Comunidades Europeas (NACE Rev.2), la cual reconoce y registra a estos servicios informáticos y al software como una subcategoría:

- i. División 62: programación de dispositivos inteligentes, asesorías y actividades relacionadas
- ii. División 63: actividades de servicio de información

Según Laporte et al., (2017), en muchas naciones de todo el mundo, sobre todo en aquellas que se encuentran en camino hacia el desarrollo nacional, el sector del software está dominado por micro o pequeñas empresas, es decir, de 20 empleados o menos, y por empresas jóvenes. En Tailandia, las micro empresas representan alrededor del 90% de la industria del software (Laporte, Muñoz, & Gerançon, 2017). En el Perú, de la totalidad del gremio empresarial que conforman a la industria elaboradora y constructora del software,

un 27% tienen el cartel de ser consideradas como pequeñas empresas y, un 63% son consideradas como micro empresas, por lo que entre ambas, cuentan con más del 90% de la totalidad de empresas de todo este sector, transformándolas en fuerza laboral considerable y sumamente relevante para la economía peruana (PromPerú, 2011).

Según Sánchez et al. (2017) al mencionar a las microempresas, se incluye a pequeñas áreas o departamentos que se encargan de la elaboración y/o construcción de software, así como también a proyectos de menor envergadura que se desenvuelven a la interna de compañías más grandes que, por lo general, emplean un capital humano menor a 25 individuos. Esta descripción o concepto ha logrado ser aceptado por el Organización Internacional de Normalización (ISO, por su acrónimo en lengua inglesa) a raíz del papel trascendental que representan estas organizaciones o micro empresas constructoras en el ámbito empresarial que se dedica a la elaboración y/o construcción del software.

Además del número de empleados, McFall, Wilkie, McCaffery, Lester, & Sterrit (2003), se dieron cuenta de que las preocupaciones y las prioridades que inquietan a los pequeños organismos que cuentan con 20 colaboradores o menos, son totalmente incomparables con aquellas inquietudes que pudieran presentar las organizaciones más grandes. La totalidad de empresas constructoras de software no son semejantes, al contrario, éstas alteran y se difieren unas de otras, de acuerdo con ciertos elementos tales como la capacidad organizacional, los años que lleva en el negocio, el estilo de administración y gestión empresarial, el sector o target del mercado al que apuntan, la variedad de productos o servicios que ofrecen, y la ubicación geográfica (Laporte, Alexandre, & O'Connor, 2008). Según Von Wangenheim & Richardson (2007), declararon que, estas compañías a menudo requieren enfoques diferentes debido a

modelos y objetivos de negocios mucho más concretos, la disponibilidad del recurso humano, el mostrarse solvente en recursos económicos, capacidad de gestión y procesos, las distintas discrepancias organizacionales, etc. Según Clarke & O'Connor (2012), definieron esto a manera del contexto de la situación, el cual envuelve ocho (08) factores a saber: negocios, gestión, operación, organización, tecnología, aplicación, requisitos, y personal.

1.3.2.2. Proceso de construcción del software en microempresas

El proceso de la elaboración y construcción del software involucra a todos aquellos movimientos, gestiones y etapas que sigue cualquier compañía con el fin de construir un producto o servicio del software. Sommerville (2011), considera que, el proceso constructivo y fabricante del software tiene que mantenerse en todo momento actualizado, mejorado y optimizado, de manera que permita el cumplimiento de los requerimientos actuales, tanto de los usuarios finales como de los clientes empresariales, sin distinción alguna. Consiguientemente, un modelo de los procesos de elaboración y construcción de software vienen a ser una representación resumida de otro proceso constructivo de software más amplio y es descriptivo ya que, indica cómo se debe desarrollar el software (p.121)

Según Pressman (2010), “existen tres categorías integrales y principales de los modelos de los procesos constructivos del software, a saber: evolutivo, incremental y cascada” (pág. 187). Asimismo, también hay otros tipos de modelo de procesos mucho más específicos, por ejemplo, tenemos, los modelos basados en pruebas y los modelos basados en componentes. Sin embargo, Turner & Boehm (2003), subrayaron la existencia de dos (02) categorías fundamentales para el proceso constructivo de software: ágil e impulsado por el plan, que se han considerado por tradición como oponentes: por un lado, los métodos caracterizados como ágiles que se encuentran basados en el desarrollo de naturaleza incremental e

iterativa, utilizando fases breves en su elaboración y desarrollo (p.101). La preferencia más relevante de las metodologías ágiles es conservar al usuario o cliente, de manera que se encuentre satisfactoriamente conforme, con una entrega mucho más temprana en referencia a los plazos establecidos con anterioridad, y continua de acuerdo a las funciones del producto o servicio software que ha adquirido. A pesar de que, los métodos ágiles de elaboración y construcción de software han llamado la atención de los ingenieros e investigadores de software a nivel mundial, “este aspecto sigue teniendo carencias y dificultades puesto que no se ha explotado su investigación en profundidad” (Abrahamsson, Oza, & Siponen, 2010, pág. 34). Sin embargo, el mundo tradicional de los procesos de elaboración y construcción de software, el cual se encuentra representado por el mejoramiento de los procesos y por los protectores de la ingeniería, contiene métodos basados en planes, los cuales están basados en la calidad de los dispositivos de las ingenierías de software y en la pre visibilidad de cada uno de sus procesos (Boehm & Turner, 2003).

En la praxis, la elaboración y construcción de software se encuentra plagado de muchos desafíos y muchas restricciones. Aunque existen múltiples orientaciones que buscan lograr la organización de los procesos de elaboración y construcción de software, y además de ello, existen factores diversos que además son influyentes en el proceso constructivo del software, las PYME podrían asumir una prioridad muy baja en dicho proceso constructivo del software puesto que, “estas se centran en la calidad del servicio o producto y los plazos de distribución determinados, más que en la calidad de los procesos mismos” (Clarke & O'Connor, 2013, pág. 984).

Según Clarke & O'Connor (2012), las PYME de software informan que practican una metodología de combinación para los procesos constructivos de software, esta combinación, mezcla

diversos aspectos de muchas orientaciones de elaboración y construcción de software prescritos de manera que, logren la satisfacción de sus requerimientos, teniendo en cuenta sus limitaciones, pero sin que ello resulte un menoscabo en sus actividades. En palabras más simples, dichas compañías u organizaciones carecen de un modelo previamente puesto en marcha para sus procesos de elaboración o construcción de software a manera de guía o política, prefiriendo en cambio dejar caer elementos de su modelo elegido o, desarrollar algo patentado que mejor se adapte a sus requerimientos específicos. Asimismo, los procesos de actividades relacionados a la ingeniería del software, se eligen de forma oportunista, han sido acondicionados y configurados a conveniencia, para lograr brindar algún tipo de valor bajo las limitaciones atribuidas por las circunstancias mismas. Efectivamente, “las diversas compañías vienen manteniendo en adopción diversas metodologías en cada uno de sus proyectos y decantándose por el seguimiento de un enfoque híbrido para sus procesos de elaboración, construcción y optimización del software” (Vijayasathy & Butler, 2016, pág. 87).

Hay muchas evidencias de que un gran porcentaje de las micros y muy pequeñas compañías constructoras de software, sobre todo las microempresas, no se encuentran practicando los estándares existentes ya que perciben que están orientados para medianas y grandes empresas y, la revisión bibliográfica de los artículos científicos han dejado en evidencia que el nivel de percepción negativo de estas microempresas sobre las normas y estándares internacionales del modelo de proceso son principalmente impulsadas por puntos de vista negativos de costo, documentación y burocracia (Laporte, Alexandre, & O'Connor, 2008). Según Moreno, Sánchez, Colomo, & de Amescua (2014), “las PYMES generalmente requieren de asesoramiento externo para la adopción e implementación de este tipo de normas y/o estándares de nivel internacional” (pág. 6). Como

resultado, en el año 2010, la Organización Internacional de Normalización publicó el estándar ISO/IEC29110, que aborda particularmente las exigencias del ciclo de vida de software exclusivamente para micro empresas, y el cual todavía sigue en pleno desarrollo. El proceso de implementación de este estándar internacional tiende a ser en algunas ocasiones de manera compleja, en algunas ocasiones más factible, sin embargo, aún es incipiente y su impacto en la literatura también es evidente. Consiguientemente, “viene a ser un estándar que se encuentra en proceso de reciente nacimiento, por lo que existe aún mucho trabajo por llevar a cabo” (Laporte, Muñoz, & Gerançon, 2017, pág. 95).

La presencia de un modelo en el proceso constructivo y transformativo del software en ningún momento garantiza a que dicho producto o servicio software será entregado, como producto final, en los plazos previamente establecidos, o que exhibirá aquellas particularidades técnicas que guíen a lograr rasgos distintivos de calidad en el lejano plazo o que alcanzará la satisfacción de los requerimientos y demandas del cliente o usuario final (Pressman, 2010). Por lo tanto, el proceso constructivo como tal, podría estimarse de manera que logre afianzar el cumplimiento de la totalidad de los criterios básicos de todos los procesos que lograron demostrar su relevancia en el procedimiento de una praxis de éxito mediante la ingeniería del software. Por esta razón, durante esta última década emergieron diversas metodologías o enfoques justamente para lograr la valoración y el mejoramiento de los procesos constructivos del software, teniendo en consideración la coyuntura a la que muchas micro empresas constructoras de software se encuentran sometidas.

1.3.3. Modelos de procesos de construcción de software

1.3.3.1. ISO/IEC29110

El estándar internacional ISO/IEC29110 de guías y normas de ingeniería de software ha visto la luz con el fin de ser un soporte para microempresas, de modo que éstas, puedan lograr la optimización de sus procesos y sistemas constructivos y fabricantes de software. Bajo argumento de la ISO/IEC29110, los sistemas de la información se encuentran integrados tanto por software, como por hardware. La serie ISO/IEC 29110, más específicamente las guías de gestión e ingeniería, deberían ayudar a las micro empresas en la implementación de las prácticas comprobadas, a fin de obtener ventajas como aumentar la calidad de sus servicios, aumentar la calidad de sus productos, reducir su tiempo de desarrollo y ayudarlos a desarrollar su producto dentro del presupuesto y horario.

Esta ISO es una norma que en un principio se desarrolló pensando en las microempresas desarrolladoras de software en el estado mexicano. Esta norma tiene como objetivo servir de soporte a las micro empresas al mejoramiento de su sistema o proceso de elaboración, construcción y desarrollo de software, ayudándoles en la implementación de prácticas comprobadas para obtener beneficios tales como el mejoramiento y acentuación de la calidad de sus servicios o productos, reducir su tiempo de desarrollo, el porcentaje de re trabajo y desarrollar su producto, dentro del presupuesto y tiempo establecidos (Muñoz, Mejía, & Laporte, 2018).

Entre las más relevantes características de la ISO/IEC29110 se encuentran:

- a. Proporciona un conjunto de 4 perfiles de software para que las micro empresas utilicen de acuerdo con sus objetivos: el perfil inicio, el perfil básico, el perfil de intermedia y el perfil de avanzada. El perfil básico del software es el único perfil, por el momento, para el que se puede certificar una microempresa.

- b. Provee dos categorías de proceso que son las principales, el proceso de administración y de gestión de los proyectos de software y el proceso de desplegar la ejecución del software como tal mediante la implementación.
- c. Provee de una guía que puede ser utilizado con el fin del establecimiento de procesos en microempresas haciendo uso de cualesquiera enfoque o metodología que tengan previamente como ya establecida en una organización.
- d. Provee un grupo de mecanismos en el proceso, entre los cuales destacan los distintos roles, las distintas tareas, las diversas actividades, los objetivos organizacionales y los productos que se llegan a desarrollar fruto del trabajo.

Según Laporte, Muñoz, & Gerançon (2017), el perfil básico del software se compone de dos procesos:

- a. **El proceso de gestión del proyecto:** Su objetivo es determinar y lograr la implantación de las diferentes tareas relacionadas con la gestión de un proyecto de manera sistemática, de modo que las metas del proyecto se completen con una calidad idónea, en el tiempo estipulado y teniendo en consideración los costos que se esperaban. Tiene cuatro (04) actividades: la planificación de los proyectos, la ejecución del plan de los proyectos, el control y el registro evaluativo de los proyectos y, en último lugar, el cierre de los proyectos mismos.
- b. **El proceso de implementación de software:** asume como objetivo realizar de manera sistemática las actividades relacionadas con el análisis, con el diseño, con la elaboración, las pruebas y la integración, de acuerdo con

los requisitos especificados de los productos o servicios de software, ya sean estos de naturaleza modificado o totalmente nuevos. Tiene seis (06) actividades a considerar: el inicio, el análisis, el diseño, la construcción, la integración y prueba y, finalmente, el reparto y entrega del producto o servicio.

Procedimiento para la adopción de la serie ISO/IEC 29110

Según Muñoz et al. (2018), para llevar a cabo la adopción, implementación y revisión de un perfil básico de software de ISO/IEC29110 por parte de los equipos encargados del proceso constructivo del software, se debe de seguir un método de seis pasos:

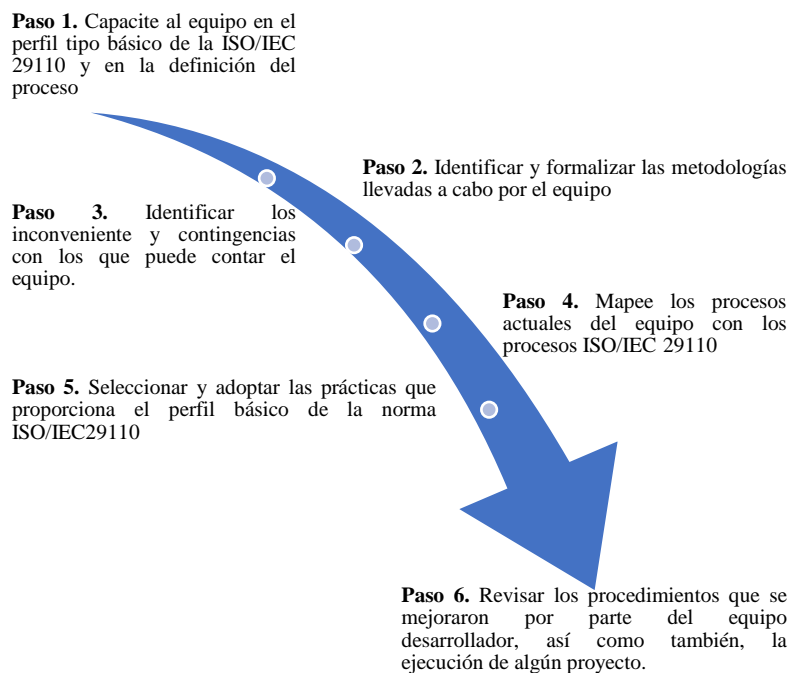


Figura 4, Pasos para adopción del software perfil básico ISO/IEC29110.

Fuente, Muñoz et al. (2018)

- a. **Paso 1.** Capacitar al equipo en el perfil básico de la ISO/IEC29110 y en la definición del proceso realizando un conjunto de talleres y seminarios de la siguiente manera: dos sesiones, un total de 15 horas, centrado en la ISO/IEC

29110 en el que se presentan y discuten los conceptos, la estructura, los procesos del software en perfil básico de ISO/IEC29110 y algunos ejemplos de su implementación y dos sesiones, por un total de 15 horas, centradas en la definición del proceso en el que los elementos del proceso, identificación de las mejores prácticas de procesos constructivos y la formalización de sus procesos característicos. Por lo tanto, se puede conocer el uso del estándar y la descripción del proceso.

- b. **Paso 2.** Identificar y formalizar las metodologías efectuadas por el equipo en el despliegue de sus procedimientos actuales. Este segundo paso se enfoca en materializar aquellas prácticas que deben desarrollarse para los procesos, tanto de la gestión de los proyectos concretamente, como el proceso de la ejecución del software como tal.
- c. **Paso 3.** Identificar los inconveniente y contingencias con los que puede contar el equipo. Este paso se centra en identificar los problemas que tiene el equipo con su forma real de trabajo, en relación con la gestión y administración del proyecto y el proceso de implantación del proyecto.
- d. **Paso 4.** Los procesos actuales del equipo de trabajo se mapean con los procesos ISO/IEC 29110, para realizar el mapeo entre el proceso actual del equipo y el perfil básico de la ISO/IEC29110. Este paso se desarrolla tanto, ya sea para los procesos de administración y gestión de los proyectos, como también para los procesos de la ejecución del software como tal.
- e. **Paso 5.** Seleccionar y adoptar las prácticas que proporciona el perfil básico de la norma ISO/IEC29110, de manera que se logre el mejoramiento de los procedimientos del equipo desarrollador, teniendo con

consideración los inconvenientes y contingencias identificadas en el paso 3.

- f. **Paso 6.** Revisar aquellos procedimientos que se lograron mejorar por parte del equipo desarrollador, así como también, la ejecución de algún tipo de proyecto. Asimismo, se hacen de conocimiento las no conformidades en lo que respecta a esta norma como tal.

1.3.3.2. ISO 12207

En 1987, la Comisión Electrotécnica Internacional (IEC) y la ISO establecieron un Comité Técnico Conjunto acerca de Tecnología de la Información. El alcance del Comité Técnico Conjunto es "Estandarización en los diversos campos de los sistemas y equipos de tecnologías informáticas (incluidos los sistemas de microprocesador)".

Según International Organization for Standardization – ISO (2008), en junio de 1989, el Comité Técnico Conjunto acerca de Tecnología de la Información inició la creación de un estándar internacional, ISO/IEC 12207, procesos de ciclo de vida del software para satisfacer una necesidad fuerte. Durante la etapa de la industria “artesanal” a fines de la década de 1970, el software se ha establecido como una parte total de muchas áreas científicas y comerciales. Pero, los ambientes para desarrollar y administrar software han proliferado por escasez de un lineamiento único para administrar y diseñar software. El Estándar Internacional llena esa necesidad crítica al instaurar un lineamiento único que podría ser usado por los expertos del software para administrar un software de ingeniería. Además, un marco tan uniforme promovería el intercambio internacional de bienes y prestaciones de software.

Fue publicado en 1995 y es un estándar internacional que envuelve el ciclo de la vida del software, partiendo desde sus orígenes

y comienzo hasta la etapa concluyente de su vida útil. El estándar se utiliza como referencia en varios países, incluido Perú (cuyo estándar equivalente es NTP-ISO/IEC12207). Este estándar de carácter internacional, constituye una arquitectura de muy elevado nivel para el ciclo de la vida de un software, el cual parte desde sus orígenes y comienzos hasta la discontinuidad y tiene como objetivo ayudar a aquellos equipos que se encuentran implicados con la elaboración y construcción del software, de modo que puedan precisar cada uno de los roles que deben de manejar y, por lo tanto, lograr suministrar a las instituciones empresariales que lo usan, una mayor comprensión de las actividades que se realizarán en los procedimientos y operaciones relacionadas con el software (ISO, 2017).

La ISO/IEC/IEEE Std.12207 proporciona un marco que incorpora todo el espectro de procesos del ciclo de vida del software. Los Objetivos de la norma 12207-2008 se pueden resumir así:

- a. Establecer un modelo reconocido internacionalmente de procesos comunes del ciclo de vida del software que pueda ser referenciado por la industria del software en todo el mundo con la finalidad de que exista una comunicación entre adquirentes, proveedores y otras partes interesadas.
- b. Servir a todos los participantes en un ciclo de vida del software: adquirentes de software y productos de sistemas, proveedores, desarrolladores, personal de operación y mantenimiento, gerentes, personal de SQA y usuarios de productos.
- c. Promover el entendimiento entre las partes comerciales mediante la aplicación de procesos, actividades y tareas comúnmente reconocidos.

El estándar es voluntario; o sea, en sí mismo no exige obligaciones para ninguno de seguirlo. Pero, podría ser impuesta por

una entidad mediante una directiva de lineamientos internos o por partes individuales mediante acuerdos contractuales. El estándar está diseñado para ser utilizado por una o más partes como base de un acuerdo o de forma autoimpuesta.

Tiene como fin ofrecer un grupo establecido de procedimientos para ayudar a comunicar a los clientes, distribuidores y el grupo de interés en el ciclo de vida de software.

Considerado como el estándar internacional que instaure lineamientos para todos, sobre los procedimientos del ciclo de la vida del software. Incluye procedimientos, acciones y trabajos a ser aplicados a lo largo de la obtención de un bien o prestación de software también durante los procesos de suministrar, desarrollar, operar, mantener y retirar los bienes de software.

Según Singh (1996), se agrupan en tres amplias clases: principales; de apoyo; y organizativos. Ver Figura 1. Los procesos principales son los motores principales en el ciclo de la vida; son de adquisición, de suministro, del desarrollo, de la operación y del mantenimiento respectivo. Comúnmente se denomina a los procesos de apoyo, como documentación, la gestión de la configuración, la garantía de la calidad, la revisión conjunta, auditoría, la verificación, la validación y la resolución de los problemas. Un proceso de soporte soporta otro proceso en el desempeño de una función especializada. Los procesos organizativos son administración, infraestructura, mejora y capacitación. Una organización puede emplear un proceso organizacional para instaurar, manejar y optimizar un proceso de ciclo de la vida.

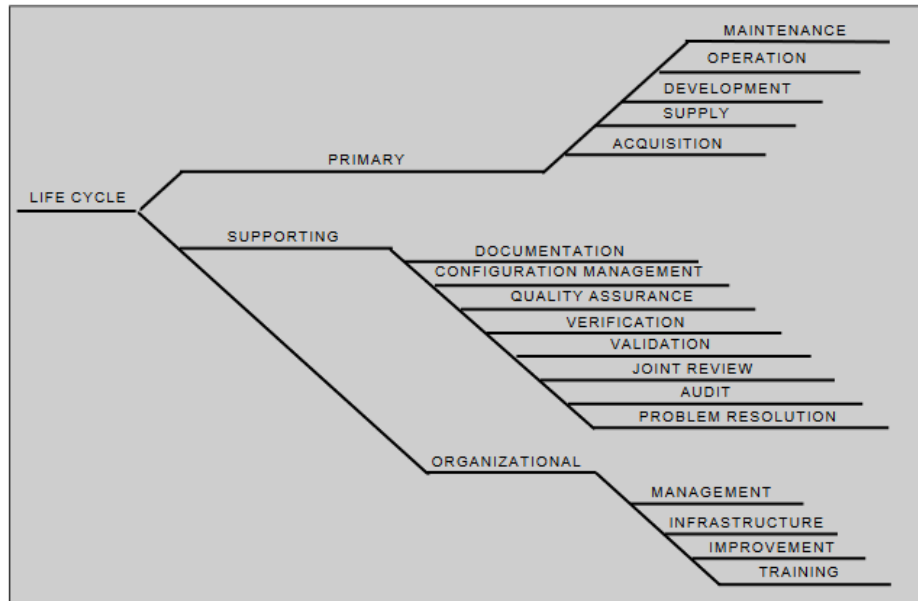


Figura 5, Procesos del ciclo de la vida del Software ISO12207.

Fuente, Singh (1996)

1.3.3.3. CMMI-DEV

La unificación de sistemas modelos de madurez de capacidades (CMMI por su acrónimo en lengua inglesa) es un modelo de procesos el cual tiene consigo todas aquellas buenas prácticas de manufactura en la industria que se encarga de la elaboración, construcción, desarrollo, adquisición, mantenimiento, operación e implementación de todos aquellos productos y servicios de software (CMMI, 2010). Se enfoca en el mejoramiento de los procesos, logrando proveer a todas organizaciones de aquellos principios que son fundamentales con el fin de lograr un proceso que sea seguro y efectivo y, que además provea a modo de modelo-guía para la consecución de la calidad en todos sus procedimientos anteriormente mencionados (Altwies & Preston, 2017). El CMMI tuvo su origen gracias a la elaboración por parte del Instituto de Ingeniería de Software (SEI por su acrónimo en lengua inglesa de Software Engineering Institute), el cual es un instituto dedicado a la investigación y desarrollo de software. El principal objetivo que busca este modelo de procesos es que, todo producto o servicio de software

cuenta con muy elevados estándares de calidad, y los cuales puedan ser cuantificables y medibles, de manera que el producto final goce de excelente reputación (Conrad, Misener, & Feldman, 2016).

“El modelo CMMI-DEV (Capability Maturity Model Integration for Development) está organizado en 5 niveles de madurez, en una secuencia predeterminada de mejora que permite la visibilidad sobre los procesos y los resultados técnicos y gerenciales del proyecto de software” (Sánchez & Viera, 2019, pág. 135). Cada nivel consta de un grupo de áreas de proceso asociadas, con objetivos y prácticas que cumplen y actúan como referencia para lograr niveles de calidad adecuados, y cada nivel proporciona la base para el siguiente. El modelo utiliza un lenguaje común y estandarizado y está compuesto por las mejores prácticas pertenecientes a las actividades de desarrollo y mantenimiento de sistemas, cubriendo todo el ciclo de vida del producto desde la concepción hasta la entrega, como se presenta en la Figura 6.

Los niveles de madurez brindan una manera de controlar y estructurar el desempeño de la organización dentro de una determinada disciplina o conjunto de disciplinas y son etapas evolutivas bien definidas, donde cada nivel establece una parte importante del proceso.

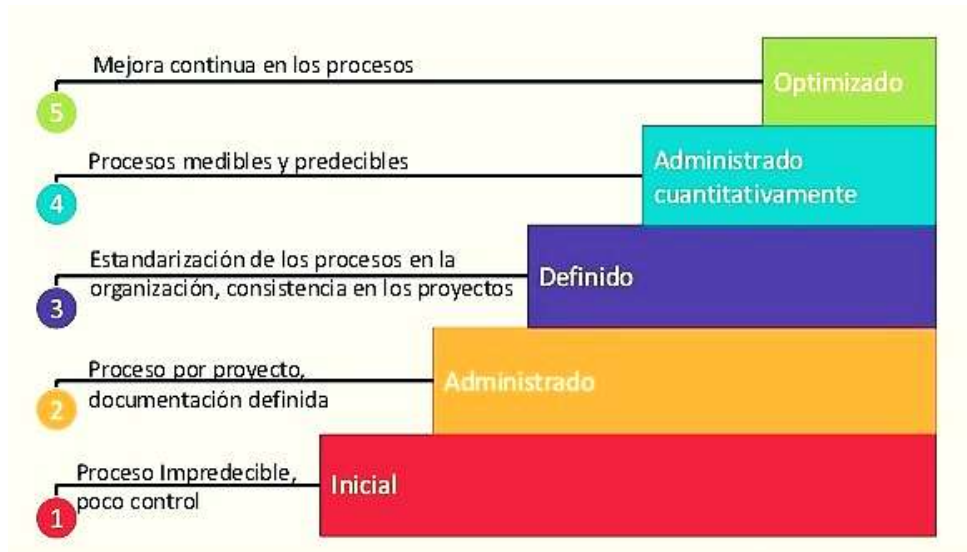


Figura 6, Niveles de madurez de CMMI.

Fuente, Sánchez & Viera (2019)

Nivel 1 - Inicial - donde los procesos son informales y el éxito esta atribuido a la competencia y ardua labor del grupo y no de la utilización de procesos probados, por lo tanto, este nivel no tiene asignaciones y no es evaluado por el modelo; Nivel 2 - Gestionado - asegura la gestión de los proyectos, los requisitos y toda su planificación y ejecución; Nivel 3 - Definido - sus procesos están bien caracterizados, comprendidos y descritos; Nivel 4 - Gestionado cuantitativamente - tiene objetivos cuantitativos para la calidad y el rendimiento del proceso; Nivel 5 - Optimizado - donde los procesos se mejoran continuamente.

Arboleda, Paz & Casallas (2013) elaboraron un modelo para el proceso completo de Planeación de Proyectos, Monitoreo y Control de Proyectos y Medición y Análisis, el cual buscó implementar el Modelo CMMI-DEV:

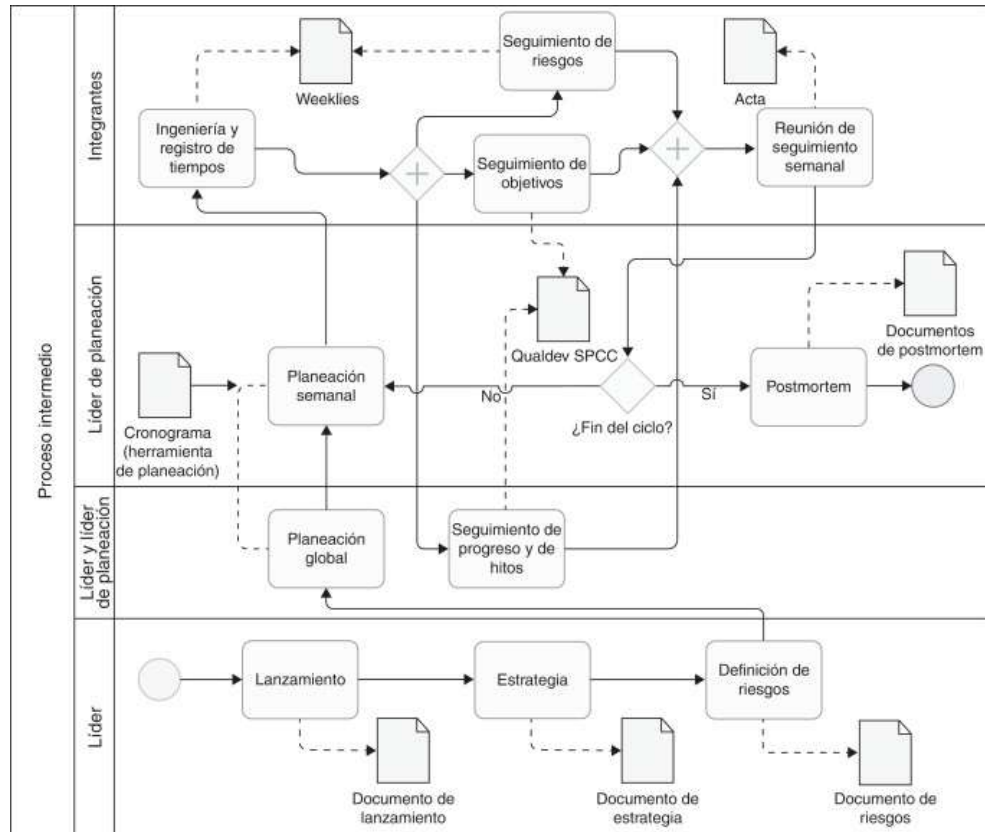


Figura 7, Modelo para la construcción de software basado en CMMI-DEV

Fuente, Arboleda et al. (2013)

1.3.3.4. MOPROSOFT

El Modelo de Procesos para la Industria de Software (MoProSoft) en el Estado de México, tuvo su inicio en el 2003 con el fin de promover la estandarización en todas las organizaciones mexicanas constructoras y fabricadoras de software mediante la presentación de excelentes prácticas de manufactura en lo que respecta a la gestión y administración de la ingeniería del software en sus procesos y, el cual ha sido ascendido al nivel de norma en el año 2005, por lo que desde ese preciso momento se convierte en la norma mexicana NMXI-059/02-NYCE-2005. MoProSoft busca suministrar un modelo que se encuentre fundamentado en la adopción de excelentes prácticas de manufactura de carácter internacional, de modo que pueda lograr estandarizar aquellos procesos de elaboración y construcción del software en el estado mexicano teniendo en

consideración que, “esta industria del software en tierras aztecas se encuentra conformada en su gran mayoría pequeñas y medianas compañías” (Valtierra, Muñoz, & Mejía, 2013, pág. 226). Este modelo busca ser un soporte para las compañías en la unificación de la mejora continua, en la evaluación de la eficacia de la misma y, además, en la estandarización de sus prácticas mediante sus tres categorías a los que hace referencia: a. Gerencia de Alta Dirección, b. Gerencia de Gestión y, c. Gerencia de Operación.

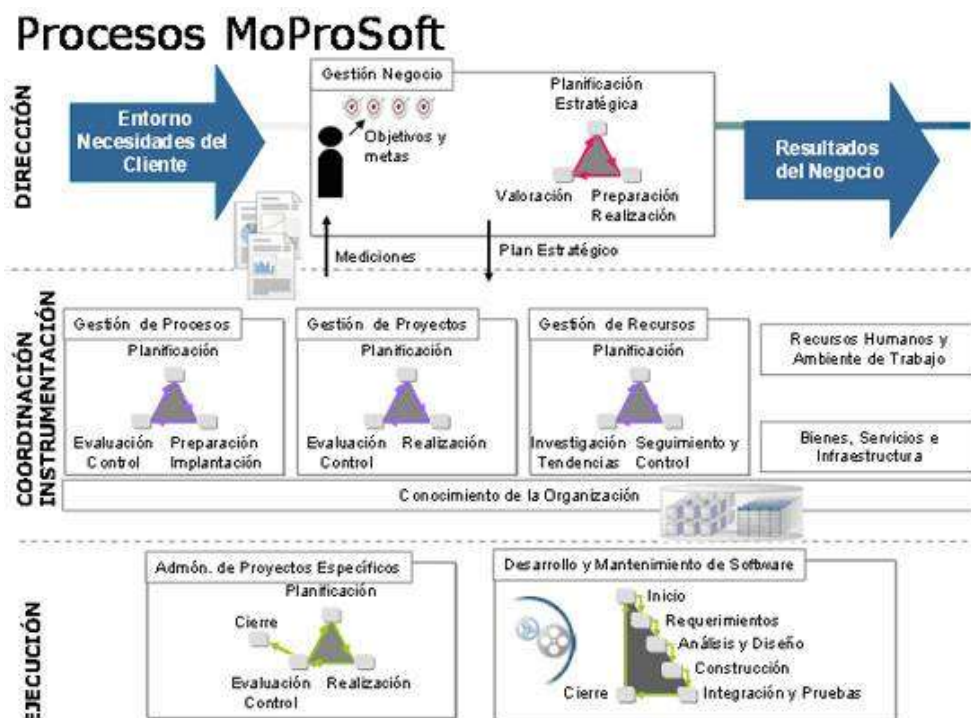


Figura 8, Categorías y procesos de MoProSoft.

Fuente, Oktaba (2005)

1.3.3.5. COMPETISOFT

El modelo de referencia del proceso COMPETISOFT se encuentra basado y referenciado en base al modelo MoProSoft, el cual ha sido definido anteriormente. Los procesos se organizan en tres categorías: alta dirección, gestión y operación; tal y como el modelo de referencia en el cual se basa. La intención de este modelo es la de suministrar procesos, los cuales deben de ser exclusivos para cada uno de los grupos funcionales de la compañía fabricante y

constructora del software. Las mejoras y mejoras propuestas en el Modelo de Referencia del Proceso COMPETISOFT son las siguientes: el proceso de la administración y gestión empresarial, el proceso de la gestión de cada uno de los procesos mismos, el proceso de la administración y gestión de los proyectos y proceso de la gestión de los recursos (Pino, García, Piattini, & Oktaba, 2016).

Villarroel (2016) desarrolló un modelo para la mejora del proceso de construcción de software basado en el modelo Competisoft, el cual se ejemplifica en la Figura 9.

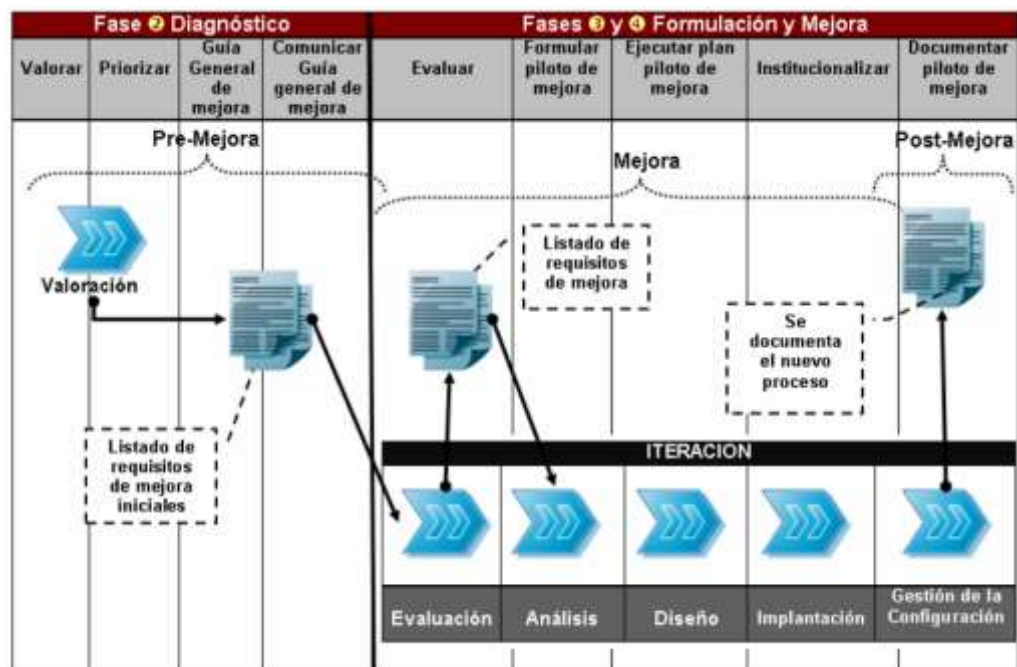


Figura 9, Modelo para la construcción de software basado en el Modelo Competisoft

Fuente, Arboleda et al. (2013).

1.4. Formulación del Problema.

¿Cómo mejorar la calidad de los procesos de construcción de software en las microempresas desarrolladoras de software en el Perú?

1.5. Justificación e importancia del estudio.

Justificación Teórica

Según Ñaupas et al. (2014), la justificación teórica es llevada a cabo cuando el fin de la investigación es la generación de reflexiones, de cuestionamientos y de debates académicos acerca de una base teórica que muestra un conocimiento ya preexistente, logrando de esta manera, por ejemplo, la confrontación de una teoría, la verificación y comprobación de los resultados, hacer gnoseología del conocimiento preexistente o en ocasiones, en las que se buscan exponer la solución de algún o algunos modelos.

El presente proyecto de investigación tiene la firme intención de contribuir con el conocimiento existente sobre importancia que tiene un correcto proceso de construcción de software en las microempresas peruanas o en pequeñas organizaciones de hasta 25 trabajadores, en las que, por medio de este modelo buscan agenciarse de resultados que evidencien su interrelación con la calidad del proceso de construcción de software orientado hacia aquellas organizaciones que cuentan con un número pequeño de colaboradores. Por ello, a través de este proyecto de investigación trabajado, se procuró el desarrollo de un modelo que mejora los procesos para la construcción de software, de modo que estas pequeñas empresas u organizaciones puedan mantenerse en el negocio mediante la construcción de productos y servicios de software con alta calidad, aún si éstas cuentan con recursos limitados.

Justificación Metodológica

Según Ñaupas et al. (2014), la justificación metodológica es llevada a cabo cuando se señala la utilización de algunos instrumentos o técnicas para la investigación y el discernimiento de manera puntual, de manera que puedan aprovecharse también para futuras investigaciones que traten de solucionar problemáticas de carácter similar.

El presente proyecto de investigación manipulará como método de recolección de datos la encuesta a la muestra que, en este caso, son las micro empresas que pertenecen al rubro de la construcción de software de la ciudad de Lima Metropolitana. Asimismo, el instrumento a utilizar será el cuestionario, en el cual la muestra responderá de manera escrita las interrogantes de dicho documento y que se encuentran ligadas a los diversos modelos de procesos existentes para

la elaboración, fabricación y optimización del software y, metodologías de elaboración y construcción del software que emplean tales compañías.

Justificación Social

Según Ñaupas et al. (2014), la justificación social responde a las preguntas citadas a continuación: ¿Cuál es su trascendencia para la colectividad?, ¿quiénes serán beneficiados con dichos resultados y las conclusiones del proyecto de investigación?, ¿de qué modo? Resumidamente, ¿Tiene algún tipo de proyección social?

El presente proyecto de investigación, desde la perspectiva del panorama social, tiene mayor aprovechamiento y ventajas para las gerencias y los empresarios, quienes son los ejes principales de la toma de decisiones empresariales, la cual se ejecutaría a través de un modelo de procesos para la elaboración y construcción del software. Asimismo, con el presente proyecto de investigación se persigue como fin el de dar a entender a la sociedad sobre la relevancia de que las compañías cuenten con modelos de procesos para la elaboración, construcción y optimización del software, que es un recurso estratégico que por seguro orienta al negocio y permite que los empresarios puedan mantenerse en el negocio mediante la elaboración y construcción de productos y servicios de software con calidad, aún si sus compañías cuentan con recursos limitados.

1.6. Hipótesis.

Con la implementación de un modelo de procesos adaptado se podrá mejorar la calidad de los procesos para la construcción de software en las microempresas desarrolladoras de software en el Perú.

1.7. Objetivos.

1.7.1. Objetivo general.

Desarrollar un modelo de procesos para la construcción de software de calidad en microempresas peruanas desarrolladoras de software.

1.7.2. Objetivos específicos.

- a. Caracterizar los modelos actuales de procesos para la construcción de software de calidad orientados a microempresas.
- b. Analizar los principales modelos de procesos para la construcción de software de calidad existentes.
- c. Diseñar un modelo de procesos para la construcción de software de calidad adecuado para una microempresa peruana desarrolladora de software.
- d. Validar mediante juicio de expertos la propuesta del modelo de procesos para la construcción de software de calidad para microempresas peruanas.

II. MÉTODO

2.1. Tipo y Diseño de Investigación.

Tipo de Investigación

El presente proyecto de investigación tuvo como tipo de investigación la tecnología aplicada, puesto que llevó a la construcción de un modelo de procesos para la construcción y elaboración del software, fomentando la búsqueda programada y organizada de la innovación tecnológica. Asimismo, fue de tipo descriptivo puesto que “permitió desplegar un análisis de las características de los objetos, tanto así que, a posteriori estos logren ser de usanza y manifestados en diversos esquemas y diagramas” (Bernal, 2010, pág. 117).

Diseño de Investigación

Por su naturaleza, el estudio correspondió al tipo de diseño experimental, ya que en el mismo se analizó “aquel efecto que se produce por la manipulación o acción de una o más variables de carácter independiente, sobre una o más variables de carácter dependientes, en un solo grupo, no habrá grupo de control” (Bernal, 2010, pág. 128).

2.2. Población y muestra.

Población

Según Ñaupás et al. (2014) definen a una población como cualquier grupo de individuos, elementos o cualquier otro constituyente, de los cuales se pretenden investigar y analizar sus particularidades, o una de estas, y a quienes serán referidas y validadas las conclusiones a las que se lleguen a obtener en una investigación.

La población considerada para este presente proyecto de investigación estuvo representada por todos aquellos modelos que se obtuvieron en la revisión sistemática de los papers, concernientes a los modelos de procesos para la elaboración y construcción del software más utilizados actualmente para estos procesos constructivos del software que busca lograr la calidad, los cuales son cinco (05). Se detallan a continuación:

Tabla 1

Población de estudio.

Nº	Modelo	Proveedor	Tipo Empresa	Antecedente
1	ISO 29910	ISO	Pequeñas organizaciones	Muñoz et al., 2020
2	ISO 12207	ISO	Pequeñas y medianas organizaciones	Laporte et al., 2017
3	CMMI	ISACA	Empresas en General	Espejo et al., 2016
4	MOPROSOFT	ISO	Pequeñas organizaciones	Sánchez et al., 2017
5	COMPETISOF	SEI	Pequeñas y medianas organizaciones	Suárez & León, 2019

Fuente: elaboración propia

Muestra

Ñaupas et al. (2014) definen que, el muestreo por conveniencia viene a ser un tipo de muestreo no aleatorio o muestreo no probabilístico, en el cual los individuos, elementos o cualquier otro constituyente de una población objetivo, que cumplen con algunos criterio, principios o particularidades prácticas, tales como la disponibilidad en un tiempo o plazo establecido, la facilidad de acceso a ellos, la cercanía geográfica, la voluntad de participación, alguna característica propia que se pretende investigar; son incluidos para poder llevar a cabo la investigación. También se refiere a los temas de investigación de la población de fácil acceso para el investigador.

Mediante el muestreo no probabilístico de tipo intencional o por conveniencia, y amparándose en este principio, se pudo establecer que la muestra estuvo constituida por aquellos modelos que se enfocan plenamente en pequeñas organizaciones, por lo que, según la Tabla 1, la norma ISO 29110 y el modelo Moprosoft son los que se adecúan a este criterio de selección no aleatoria.

Tabla 2

Muestra de estudio.

N°	Modelo	Proveedor	Tipo Empresa	Antecedente
1	ISO 29910	ISO	Pequeñas organizaciones	Muñoz et al., 2020
4	MOPROSOFT	ISO	Pequeñas organizaciones	Sánchez et al., 2017

Fuente: elaboración propia

Esta norma y este modelo son los que se analizaron y se utilizaron como fundamento para el modelo del presente proyecto.

Asimismo, para la evaluación del modelo propuesto, se evaluaron seis (06) proyectos de construcción de software a los cuales se les hicieron las mediciones pertinentes en cuanto a los indicadores especificados en la Tabla 5. Dichos proyectos constructivos se desarrollaron por Sistema Inteligente ERP SAC y corresponden a proyectos construidos en meses de la siguiente manera:

Tabla 3

Sub grupo de la muestra.

	J20	J20	A20	S20	O20	N20	D20	E21	F21	M21	A21	M21
Pre	x	x	x	x	x	x						
Test												
Post							x	x	x	x	x	x
Test												

Fuente: elaboración propia

2.3. Variables, Operacionalización.

Variable Independiente: Modelo de Procesos para la construcción de software

Variable Dependiente: Calidad de los procesos para la construcción de software

Tabla 4

Operacionalización de la variable independiente.

Variables	Dimensión	Indicador	Ítem	Técnica e instrumentos de recolección de datos
VARIABLE INDEPENDIENTE: Modelo de Procesos para la construcción de software.	Aceptación de Modelo	Nivel de Claridad	$N_{Claridad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	T: Juicio de Expertos I: Ficha de Juicio de Expertos
		Nivel de Objetividad	$N_{Objetividad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Actualidad	$N_{Actualidad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Organización	$N_{Organización} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Suficiencia	$N_{Suficiencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Intencionalidad	$N_{Intencionalidad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Consistencia	$N_{Consistencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Coherencia	$N_{Coherencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Metodología	$N_{Metodologia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	
		Nivel de Pertinencia	$N_{Pertinencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$	

Fuente: Elaboración propia

Tabla 5

Operacionalización de la variable dependiente.

Variables	Dimensión	Indicador	Ítem	Técnica e instrumentos de recolección de datos
<p>VARIABLE DEPENDIENTE: Calidad de los procesos para la construcción de software.</p>	Defectos por proyecto	Cantidad de defectos	$CDP = \sum (DE)$ <p>Leyenda: CDP: Cantidad de defectos en el proyecto DE: Defecto</p>	<p>T: Observación</p> <p>I: Ficha de Observación</p>
	Satisfacción del Cliente	Porcentaje de Clientes Satisfechos	$P_{CS} = \frac{(TC - CI)}{TC} * 100$ <p>Leyenda: P_{CS}: Porcentaje de Clientes Satisfechos TC: Total de Clientes del Proyecto CI: Clientes que presentan algún tipo de inconformidad</p>	<p>T: Encuesta</p> <p>I: Ficha de Encuesta vía Formulario Google</p>

2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad.

Juicio de Expertos

Según Hallowell & Gambatese (2010), el Juicio de Expertos es una técnica para la averiguación e investigación, la misma que tiene carácter interactiva y sistémica, la cual permite apoderarse del juicio de un panel de expertos, los cuales deben de tener carácter de independientes con la investigación, sobre un tema en concreto. Estos expertos serán seleccionados teniendo en consideración pautas ya definidas con total antelación, por lo que se les pedirá su participación en una intervención para dar su veredicto acerca de un plan, modelo, metodología u otro.

Para el presente proyecto, se utilizó este juicio de expertos por parte de ingenieros de sistemas que cuentan con experiencia en Ingeniería de Software, toda vez que se preservó la heterogeneidad de los autores, de modo que se aseguró que los resultados que se desarrollaron en el presente trabajo de grado cuenten con una validez y una confiabilidad. Esta técnica fue utilizada para validar la variable independiente “Modelo de Procesos para la construcción de software”.

Observación directa

Según Wildemuth (2016), la observación directa viene a ser una de las metodologías y métodos más comunes para lograr una correcta recolección de datos, además de ser también, uno de los más exigentes. Esta técnica de recolección de datos pretende que todo investigador logre convertirse en un colaborador partícipe dentro del contexto o situación a la que se pretende prestar atención. La observación directa, por lo general, precisa y requiere de cuantiosos meses y hasta años, en algunos casos, todos ellos de una labor intensa y de mucho rigor ya que, los investigadores precisan de lograr la aceptación como actores naturales de la situación, de modo que aseguren que las observaciones hechas en su recopilación de información, sean veraces y del fenómeno natural que pretender visualizar (p.209)

Para el presente proyecto, se hizo uso de la observación directa en el caso de estudio, la empresa Sistema Inteligente ERP SAC puesto que, permitió verificar este objeto de investigación en diversas situaciones particulares, la de construcción de software en seis (06) proyectos que se mencionaron anteriormente en la muestra de estudio, sin tener que lograr la alteración del ambiente en el que se desenvuelven los procesos, lo que permitió lograr datos e información válida y confiable, la cual fue relevante para el modelado del proceso de elaboración y construcción de software que se pretendía proponer, sobre todo al obtener información del grupo pre test. Esta técnica fue utilizada para validar la variable dependiente “Calidad de los procesos para la construcción de software”.

Encuesta

Según Terrell (2012) la encuesta es “un método de recopilación de información de una muestra de personas, tradicionalmente con la intención de generalizar los resultados a una población más amplia”. Las encuestas actúan como una base de datos que ofrecen una fuente crítica de conocimientos y datos para casi todos los involucrados en cualquier rama del mundo académico y/o investigativo.

Para la presente investigación, más específicamente, para la validación de los resultados obtenidos en la variable dependiente, se tuvo que desarrollar una encuesta mediante Formulario Google con el que se pudiera medir el porcentaje de clientes satisfechos para cada proyecto. Vale recalcar que se recurrió a dicha herramienta brindada por Google puesto que, por cuestión de la emergencia sanitaria no se contaba con autorización de la empresa para hacer reuniones presenciales con los clientes, sumado a que, la crisis de salud no lo permitía.

Dicho Formulario Google fue desarrollado por los autores en su totalidad y fue repartido vía WhatsApp y correo electrónico a los clientes de Sistema Inteligente ERP SAC. Tal formulario se encuentra desplegado en los anexos y alojado en el siguiente link: <https://forms.gle/seRKsrT9ZsVWtzzJ7>.

The image shows a Google Forms survey titled "Sistema Inteligente ERP". The header features a logo with four circles (two blue, two green) and the text "Sistema Inteligente". Below the header, the title "Sistema Inteligente ERP" is displayed. The survey text reads: "La siguiente encuesta es para ver su satisfacción como cliente, sobre el proyecto desarrollado con nosotros y cuanto nos recomendaría." The user is identified as "vmorenojorgelui@crece.uss.edu.pe (no compartidos)" with a "Cambiar de cuenta" link. The first question is "¿El software funciona y cumple sus expectativas?" with radio button options for "Bueno", "Regular", and "Malo". The second question is "¿Es la interfaz de nuestro software fácil de usar?".

Figura 10, Encuesta desarrollada vía Formulario Google

Fuente, elaboración propia.

Revisión Bibliográfica

Bajo la perspectiva de King, Hooper, & Wood (2011), una revisión bibliográfica se encarga de examinar fuentes académicas, entre las cuales destacan artículos de rigor académicos, revistas de carácter científicas, conferencias publicadas, libros y cualquier otra fuente que se considere sumamente relevante para una temática en específico, un área de investigación o teoría, y que al desarrollar dicha examinación, proporcione una descripción, un sumario y una evaluación crítica por parte de estas investigaciones en correcta interrelación con la problemática y temática investigativa que se plantea y se encuentra en fase investigativa (p.33).

Para el vigente proyecto de investigación, se echó mano de esta técnica de revisión bibliográfica, ya que la misma ha sido planteada con el fin de lograr proporcionar una vista completa de diversas fuentes científicas, las mismas que han sido exploradas de manera sincronizada mientras se llevaba a cabo el tema de este proyecto en particular, demostrando la manera en que encaja esta pesquisa al interior de un campo de estudio con mayor amplitud. Para ello, se examinaron bases de datos de información veraz y confiable, todo ello con carácter científico. Entre las bases de datos científicas utilizadas tenemos a IEEExplore, IOP, SciELO, Science Direct, Dialnet, EBSCO, PROQUEST, SCOPUS y VILEX.

Caso de estudio

Según Gagnon (2010), un caso de estudio se conceptualiza como un intenso análisis de cualquier contexto en particular o de cualquier individuo de una manera específica. Por ejemplo, Freud llevó a cabo una investigación del caso de diversos individuos, lo que le resultó como pilar fundamental para su posterior Teoría Psicoanalítica. Otro ejemplo es el de Piaget quien llevó a cabo estudios de casos de infantes con lo que logró pesquisar las distintas fases del desarrollo humano. En realidad, no existe un canon, ni hay quorum por parte de los expertos acerca de una manera universal o estandarizada de cómo se debe de desplegar un caso de estudio, por lo que, en algunas situaciones, podría darse la situación de echar mano de una combinación de varios métodos, formando un método mixto, usando en algunos casos, si queremos tomar como ejemplo, esta técnica de caso de estudio con algún tipo de entrevista no estructurada.

Para el presente proyecto, se hizo uso de un caso de estudio que fue la empresa Sistema Inteligente ERP SAC. Sistema Inteligente ERP SAC, es una compañía limeña de asesoría, consultoría y servicios en tecnología de información con 10 años de experiencias en la implementación de productos y herramientas informáticas originales e innovadoras a los servicios de muchas empresas y organizaciones en general. Durante este tiempo se ha especializado en desarrollar aplicaciones informáticas y dar soluciones, con enfoque en el

mejoramiento continuo de los procesos que, como organización tiene como tal, desarrollando herramientas de software que permite integrar todas las áreas de la organización y garantiza la seguridad informática, automatizando la totalidad del control documentario de sistemas de manera integral.

Sistema Inteligente ERP SAC se decanta siempre por el aprovechamiento de las tecnologías de primer nivel, por el buen desarrollo de sus productos y servicios y, sobre todo, por la calidad con la que cuentan sus productos finales. Brinda servicios los cuales cumplen con los requerimientos y con las expectativas establecidas por su clientela, de la mano con un compromiso que garantiza la responsabilidad social corporativa y, sobre todo, la conservación del medio ambiente. Por su naturaleza, Sistema Inteligente ERP SAC ofrece productos y servicios los cuales se enfocan en mejorar los procesos y operaciones que llevan a cabo sus clientes en el día a día. Entre estos servicios y producto se tiene:

- a. Consultoría Informática
- b. Desarrollo de Software
- c. Ingeniería de Redes y Sistemas
- d. Venta de Equipos de Cómputo, componentes, Suministros

2.5. Procedimiento de análisis de datos.

Según Castro, Kellison, Boyd & Kopak (2010), el procedimiento del análisis de datos fue precisado y conceptualizado a manera de un proceso que selecciona, transforma y modela los datos con el fin de revelar información, la misma que será útil con la finalidad de lograr una óptima toma de decisiones organizacionales (p.12).

2.5.1. Análisis Estadístico de Datos para la Variable Independiente

Para lograr la evaluación del modelo de procesos de elaboración y construcción del software en microempresas desarrolladoras de software en el Perú, se utilizaron los siguientes indicadores:

Nivel de Claridad: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas esta formulado con lenguaje apropiado.

$$N_{Claridad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Claridad}$: Nivel de Claridad

CE : Calificación de Experto

n = Número Total de Expertos

Nivel de Objetividad: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas está expresado en conductas observables.

$$N_{Objetividad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Objetividad}$: Nivel de Objetividad

CE : Calificación de Experto

n = Número Total de Expertos

Nivel de Actualidad: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas es adecuado al avance de la gestión del desarrollo de software.

$$N_{Actualidad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Actualidad}$: Nivel de Actualidad

CE : Calificación de Experto

n = Número Total de Expertos

Nivel de Organización: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas está elaborado con una organización lógica.

$$N_{Organización} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Organización}$: Nivel de Organización

CE : Calificación de Experto

n = Número Total de Expertos

Nivel de Suficiencia: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas comprende los aspectos en cantidad y calidad.

$$N_{Suficiencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Suficiencia}$: Nivel de Suficiencia

CE : Calificación de Experto

n = Número Total de Expertos

Nivel de Intencionalidad: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas es adecuado para valorar la gestión del desarrollo de software.

$$N_{Intencionalidad} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Intencionalidad}$: Nivel de Intencionalidad

CE: Calificación de Experto

n = Número Total de Expertos

Nivel de Consistencia: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas se encuentra basado en aspectos teóricos científicos.

$$N_{Consistencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Consistencia}$: Nivel de Consistencia

CE: Calificación de Experto

n = Número Total de Expertos

Nivel de Coherencia: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas cuenta con la coherencia necesaria entre cada uno de los pasos de dicho modelo.

$$N_{Coherencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Coherencia}$: Nivel de Coherencia

CE: Calificación de Experto

n = Número Total de Expertos

Nivel de Metodología: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas responde al propósito de la investigación.

$$N_{Metodología} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Metodología}$: Nivel de Metodología

CE : Calificación de Experto

n = Número Total de Expertos

Nivel de Pertinencia: Evalúa si el modelo de procesos para la construcción de software en microempresas peruanas es útil y adecuado para el desarrollo de software.

$$N_{Pertinencia} = \frac{(CE_1 + CE_2 + \dots + CE_n)}{n}$$

Donde:

$N_{Pertinencia}$: Nivel de Pertinencia

CE : Calificación de Experto

n = Número Total de Expertos

2.5.2. Análisis Estadístico de Datos para la Variable Dependiente

Para lograr la evaluación de la calidad de los procesos para la construcción de software en microempresas desarrolladoras de software en el Perú, se utilizaron los siguientes indicadores:

Cantidad de Defectos en el proceso de construcción del software: El cálculo de la cantidad de defectos en el proceso de elaboración y construcción del software es un indicador para la alta gerencia de toda compañía puesto que, permite establecer el número total de los errores en la elaboración del modelo de proceso para la construcción del software, con el fin de poder lograr la gestión de todos los defectos en dicho proceso de construcción a manera de mantenimiento preventivo.

$$CDP = \sum (DE)$$

Leyenda:

CDP: Cantidad de defectos en el proyecto

DE: Defecto

Porcentaje de Clientes Satisfechos: El porcentaje de los clientes satisfechos en el proceso de construcción de software es un indicador para la alta gerencia de la compañía puesto que, permite la valorización porcentual del número de clientes totales que evalúan su satisfacción en relación con diversos factores.

$$P_{CS} = \frac{(TC - CI)}{TC} * 100$$

Leyenda:

P_{CS} : Porcentaje de los Clientes Satisfechos

TC: Total de Clientes del Proyecto

CI: Clientes que presentan algún tipo de inconformidad

Asimismo, el propósito común de este procedimiento de análisis de datos fue lograr la extracción de valiosa información de los datos recopilados y lograr una óptima toma de decisiones organizacionales, la cual se basó en el análisis informativo.

Tal y como se mencionó en la Tabla 3, para el presente informe de investigación se evaluaron seis (06) proyectos de desarrollo de software a los cuales se les hicieron las mediciones respectivas en cuanto a los indicadores especificados en Tabla 5, específicamente para la variable dependiente. Dichos proyectos de software han sido desarrollados por la empresa Sistema Inteligente ERP SAC y corresponden a proyectos evaluados de la siguiente manera:

- a. Para el Pre Test:** Se evaluaron tres (03) proyectos previos al modelo propuesto. Dichos proyectos han sido elaborados durante los meses junio, julio, agosto, septiembre, octubre y noviembre del año 2020.

b. Para el Post Test: Se evaluaron tres (03) proyectos posteriores al modelo propuesto. Dichos proyectos han sido elaborados durante los meses de diciembre 2020, y enero a mayo del año 2021.

Se consideró la siguiente línea de tiempo para la evaluación de los proyectos desarrollados en la empresa Sistema Inteligente ERP SAC:

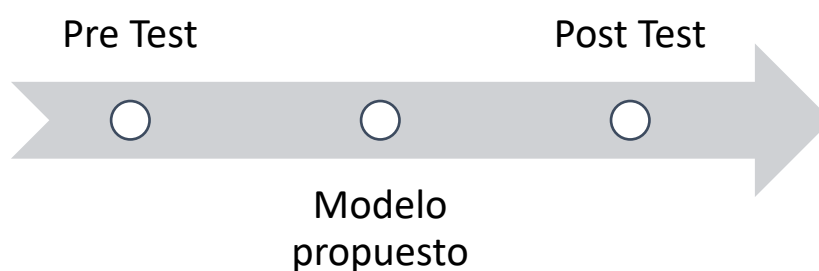


Figura 11, Línea de tiempo para la evaluación de los proyectos

Fuente, Elaboración propia.

2.6. Criterios éticos.

Según Noreña, Alcaraz, Rojas, & Rebolledo (2012), los criterios de carácter éticos que son aplicados en toda investigación de tipo cuantitativa son los siguientes:

Consentimiento Informado

Este criterio considera que, los individuos que participaron en esta investigación se mostraron en conformidad con el hecho de ser informantes y a su vez, lograron reconocer tanto sus deberes, como sus derechos.

Confidencialidad

Este criterio considera que, los individuos que participan en esta investigación fueron informados del tratamiento que se les dio a sus datos personales, por lo que la protección y seguridad de la identidad de cada uno de ellos fue importante, toda vez que fueron informantes que se encontraban prestos a participar con el fin de apoyar a la presente investigación.

Observación participante

Este criterio consideró que, los autores de este presente proyecto de investigación actuaron de una manera prudente en el desarrollo del procedimiento sistemático de recolección de datos, de tal manera que mostraron una postura y conducta totalmente éticas y responsables, con cada una de las consecuencias o los efectos que se desencadenaron de la interrelación que se pudo establecer con aquellos individuos de la población que participaron en este trabajo de grado.

2.7. Criterios de Rigor Científico.

Según Noreña et al. (2012), los criterios de rigor de carácter científico que fueron aplicados en toda investigación de tipo cuantitativa fueron los siguientes:

Credibilidad mediante el valor de la verdad y autenticidad

Este criterio se caracterizó porque obtuvo el resultado de aquellas variables que han sido estudiadas, examinadas y observadas, usando los siguientes procesos: Los resultados obtenidos fueron contemplados como verdaderos por aquellos individuos de la población que participaron del estudio, se observaron las variables en su hábitat natural de desenvolvimiento y, finalmente, se procedió a especificar la discusión de los resultados obtenidos haciendo uso del procedimiento de triangulación, el cual involucró los datos los investigadores y las teorías.

Consistencia para la replicabilidad

Este criterio se caracterizó porque adquirió el resultado que se obtuvo haciendo uso del método mixto, usando los siguientes procedimientos: La triangulación de la información logrando así fortalecer el reporte llevado a cabo en la discusión de resultados, la encuesta desarrollada mediante la usanza de un cuestionario, el mismo que fue utilizado para la recolección de datos y que fue validado por expertos, con el que se logró la autenticidad de la justificación e importancia de la investigación, al ser informaciones procedentes de fuentes primarias tales como la participación de los individuos de la población, y se detalló

coherentemente el procedimiento para la recopilación de la información, la tabulación y el posterior análisis de éstos, de manera que se logró hacer utilización de los enfoques con los que cuenta la ciencia, los cuales son: el enfoque empírico, el enfoque crítico y, el enfoque vivencial.

Confirmabilidad y neutralidad

Este criterio se caracterizó porque obtuvo resultados logrados a partir de distintas investigación, las cuales fueron veraces en la descripción, usando la siguiente metodología: Los resultados que se han podido lograr en la investigación, procedieron a ser constatados con aquellos resultados derivados de la revisión bibliográfica de papers de bases de datos con información fidedigna, confiable y veraz, todo ello con carácter científico, teniendo en cuenta que estos papers debieron de ser investigaciones del contexto local, nacional e internacional, los cuales mostraron ciertos parecidos con las variables objeto de estudio, con una antigüedad no mayor a los cinco (05) años, para luego ser declaradas varias informaciones, tales como la identificación del problema principal, el diseño de la justificación e importancia, las limitaciones del trabajo y los alcances obtenidos por los distintos autores.

Relevancia

Este criterio se caracterizó porque permitió la consecución de las metas planteadas de manera que se obtuvo un mayor y sobresaliente informe de las variables objeto de estudio, usando los siguientes procedimientos: Se llegó a la vasta comprensión de las variables estudiadas, los resultados adquiridos tuvieron correlación con la justificación e importancia de estudio.

III. RESULTADOS

3.1. Resultados en tablas y figuras.

3.1.1. Resultados de la Variable Independiente: Modelo de Procesos para la construcción de software.

3.1.1.1. Indicador Nivel de Claridad del Modelo.

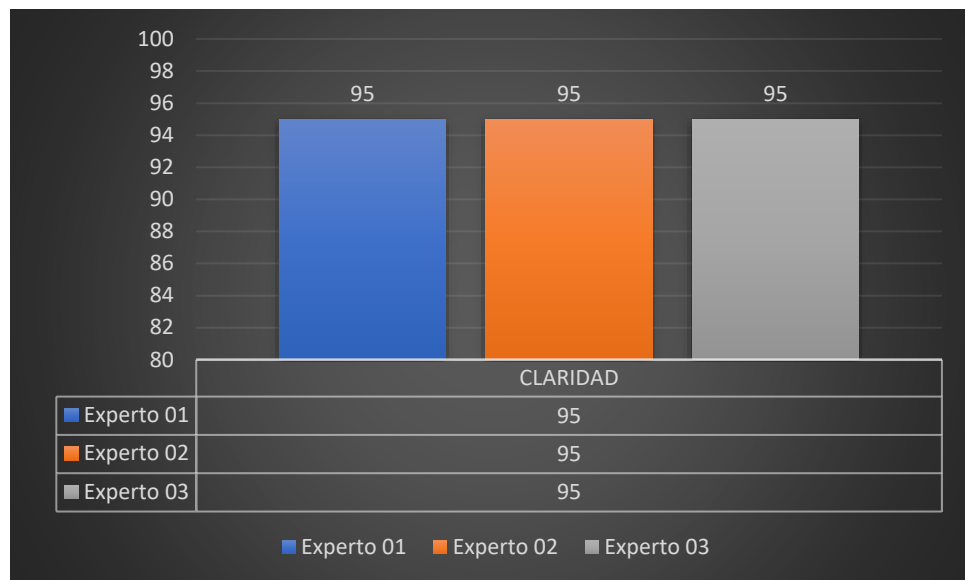


Figura 12, Indicador Nivel de Claridad del Modelo

Fuente, elaboración propia.

3.1.1.2. Indicador de Nivel de Objetividad del Modelo

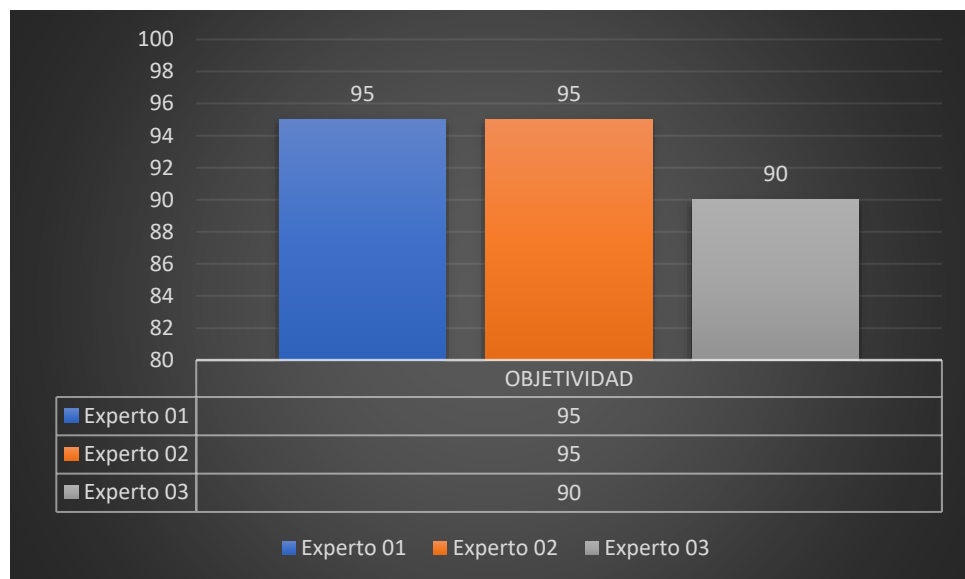


Figura 13, Indicador Nivel de Objetividad del Modelo

Fuente, elaboración propia.

3.1.1.3. Indicador de Nivel de Actualidad del Modelo

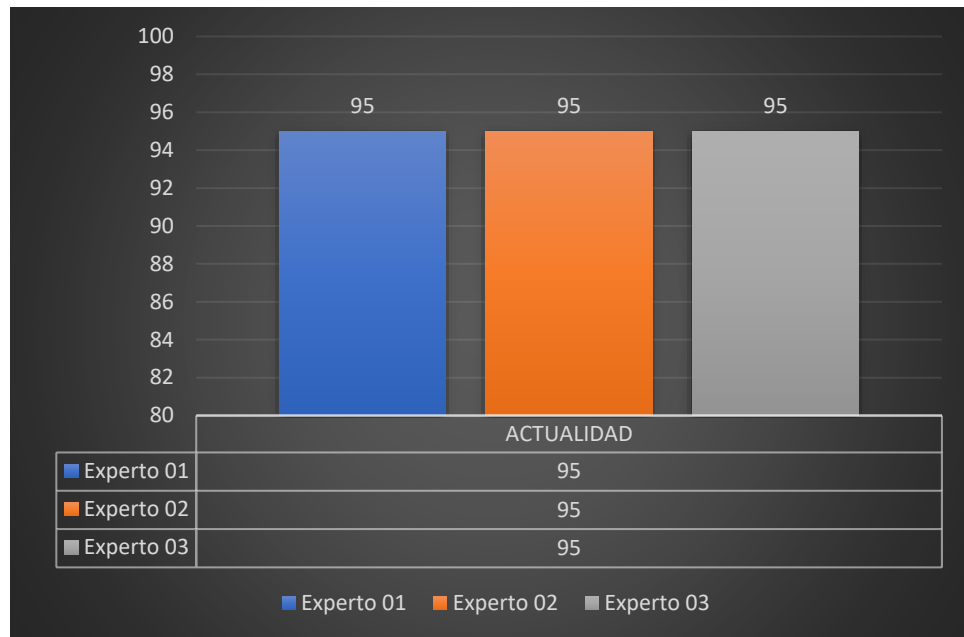


Figura 14, Indicador Nivel de Actualidad del Modelo

Fuente, elaboración propia.

3.1.1.4. Indicador de Nivel de Organización del Modelo

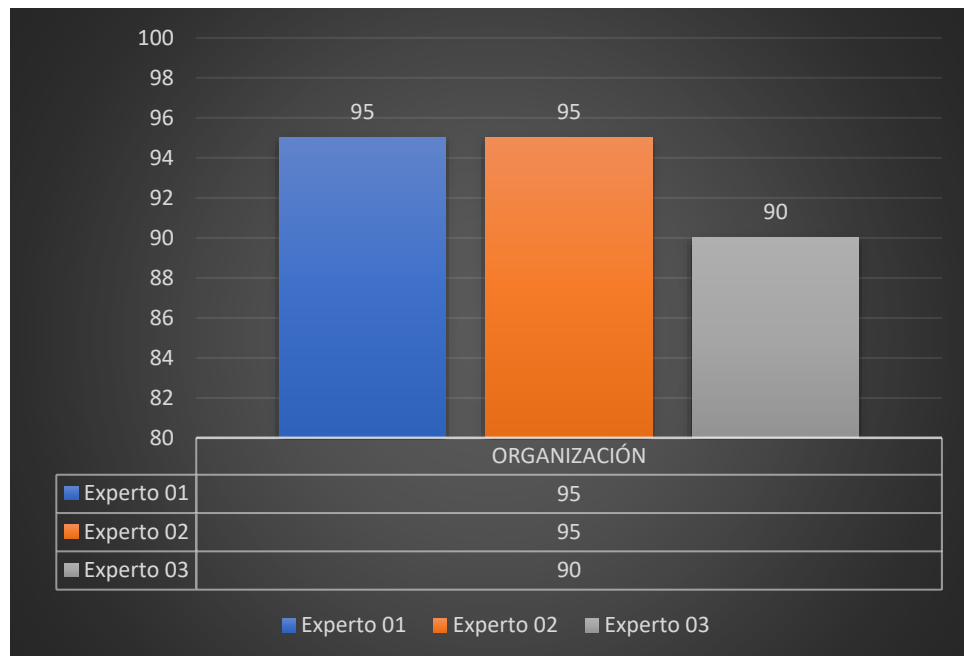


Figura 15, Indicador Nivel de Organización del Modelo

Fuente, elaboración propia.

3.1.1.5. Indicador de Nivel de Suficiencia del Modelo

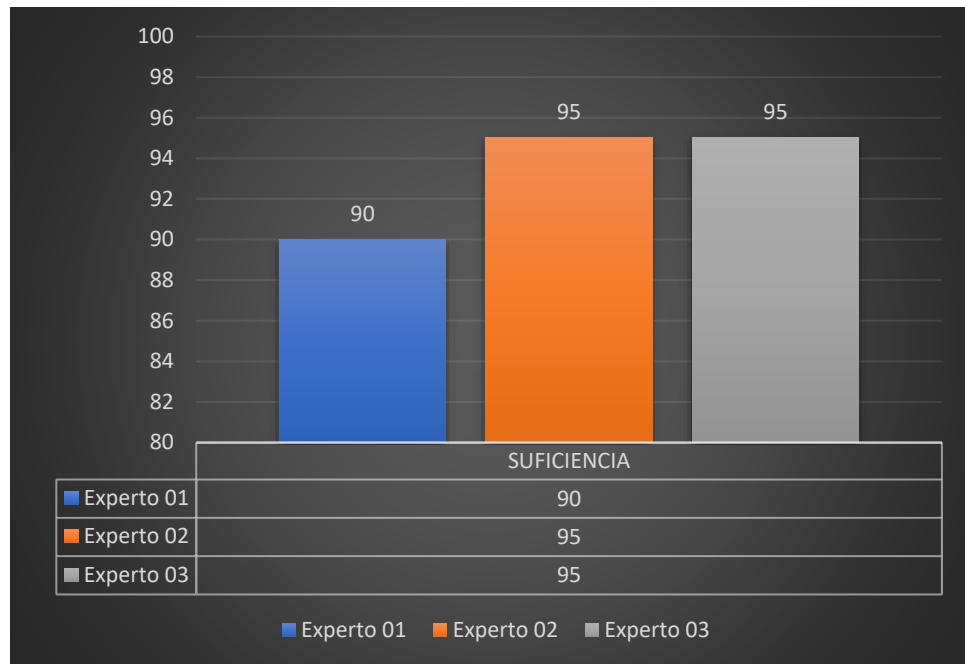


Figura 16, Indicador Nivel de Suficiencia del Modelo

Fuente, elaboración propia.

3.1.1.6. Indicador de Nivel de Intencionalidad del Modelo

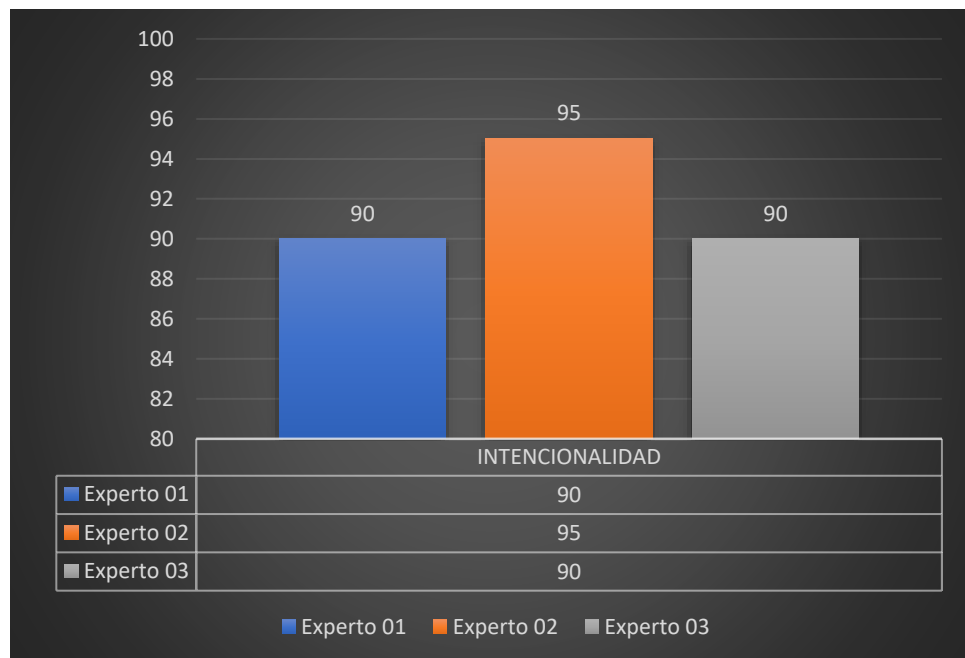


Figura 17, Indicador Nivel de Intencionalidad del Modelo

Fuente, elaboración propia.

3.1.1.7. Indicador de Nivel de Consistencia del Modelo

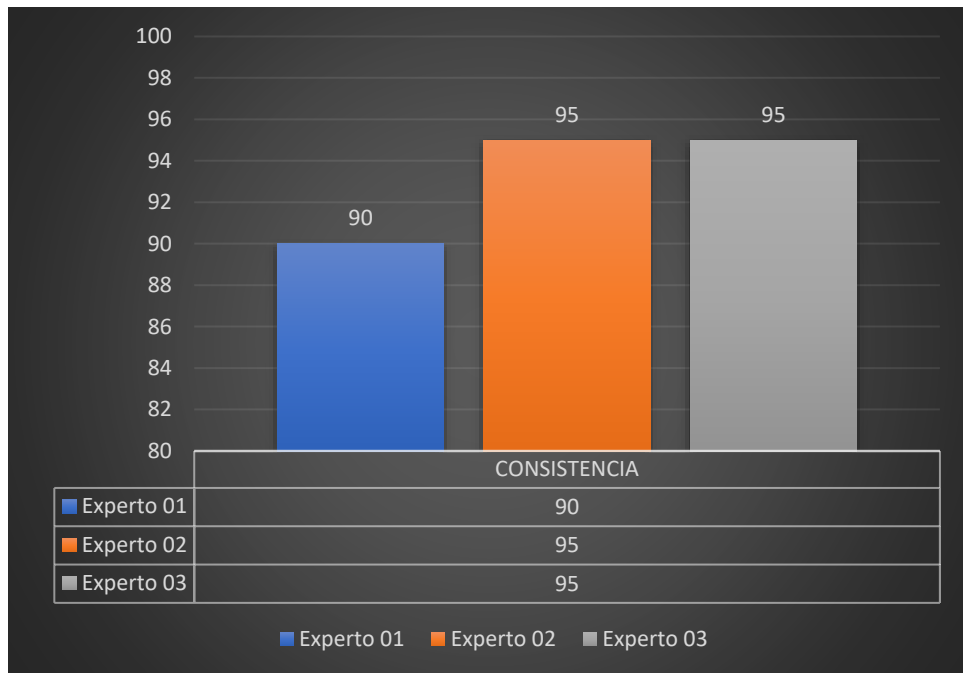


Figura 18, Indicador Nivel de Consistencia del Modelo

Fuente, elaboración propia.

3.1.1.8. Indicador de Nivel de Coherencia del Modelo

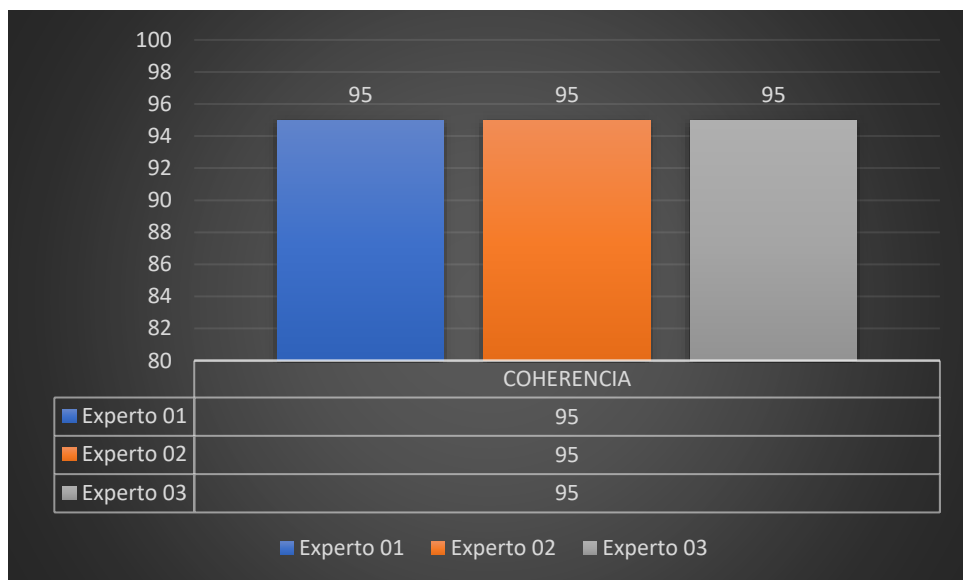


Figura 19, Indicador Nivel de Coherencia del Modelo

Fuente, elaboración propia.

3.1.1.9. Indicador de Nivel de Metodología del Modelo

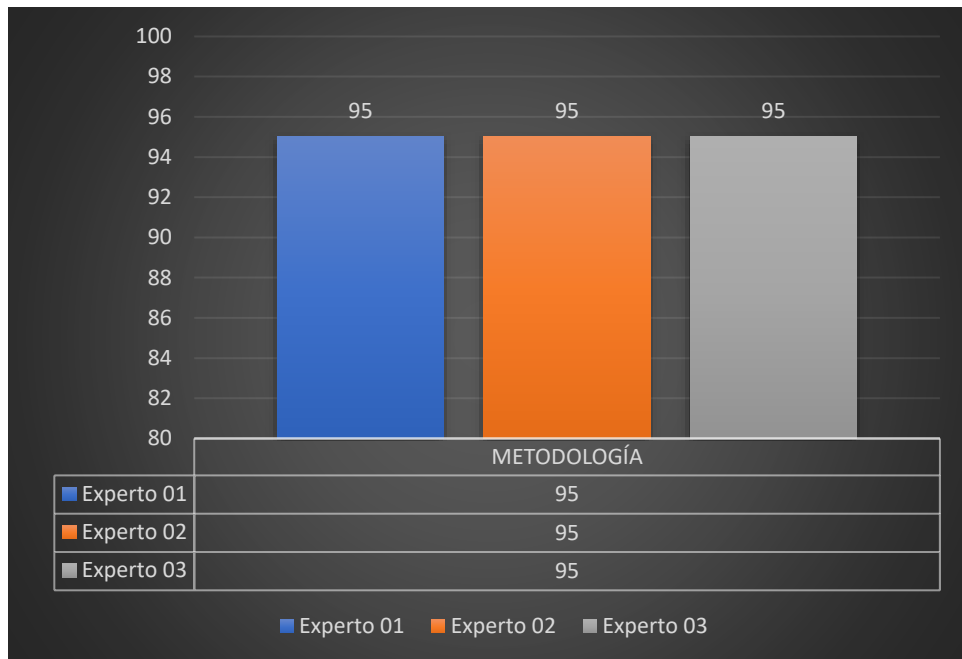


Figura 20, Indicador Nivel de Metodología del Modelo

Fuente, elaboración propia.

3.1.1.10. Indicador de Nivel de Pertinencia del Modelo

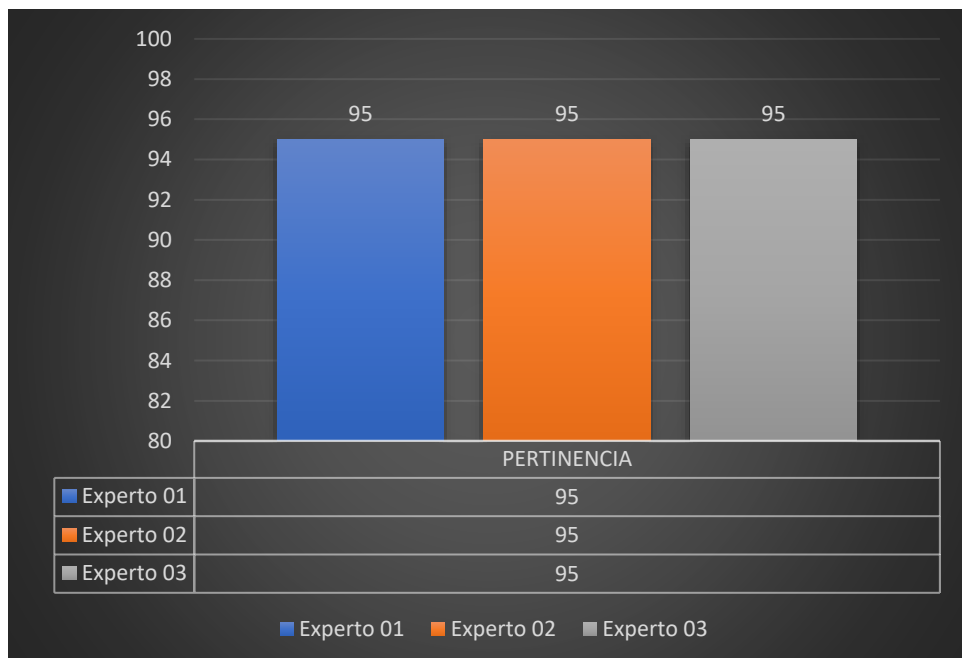


Figura 21, Indicador Nivel de Pertinencia del Modelo

Fuente, elaboración propia.

3.1.2. Resultados de la Variable Dependiente: Calidad de los procesos para la construcción de software.

3.1.2.1. Indicador Cantidad de Defectos en el proceso de construcción

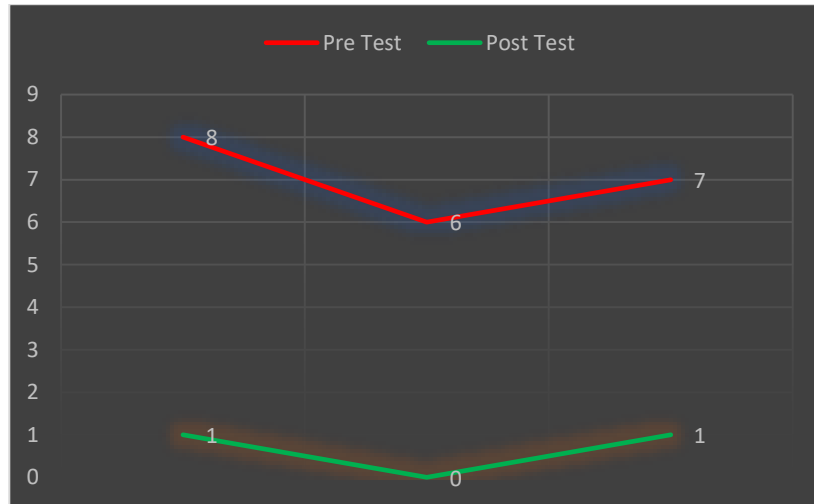


Figura 22, Indicador Defectos por proyecto.

Fuente, Elaboración propia.

Se evidenció que, los proyectos que disponían de un mayor número de defectos por proyecto se encontraban situados en el grupo pre-test con un máximo de 8 defectos en detrimento del grupo post-test donde se obtuvo un máximo de tan solamente 1 defecto por proyecto como máximo, respecto a la sumatoria de los defectos de las fases del ciclo de vida del software en todas sus etapas. Dichos defectos del grupo pre-test se evidenciaban en problemáticas relacionadas con la mala estimación de tiempos, la insuficiente administración de los riesgos, se limitaba y escatimaba el control de calidad al procesos constructivo, se diseñaba inadecuadamente, exceso de confianza hacia herramientas y tecnologías que no eran previo a su usanza, débil o carente motivación del personal y la añadidura de personal adicional a proyectos que se desfasaban en contraste con la línea de tiempo de entregas planificada.

En la Figura 23 se visualiza el promedio de cantidades que se obtienen de los defectos por proyecto correspondiente a los grupos de pre-test y post-test, donde se observa una reducción en promedio de 19 defectos por grupo

de proyectos. Se puede disminuir de 21 defectos obtenidos en el grupo pre-test a tan solo 2 defectos obtenido en el grupo post-test.

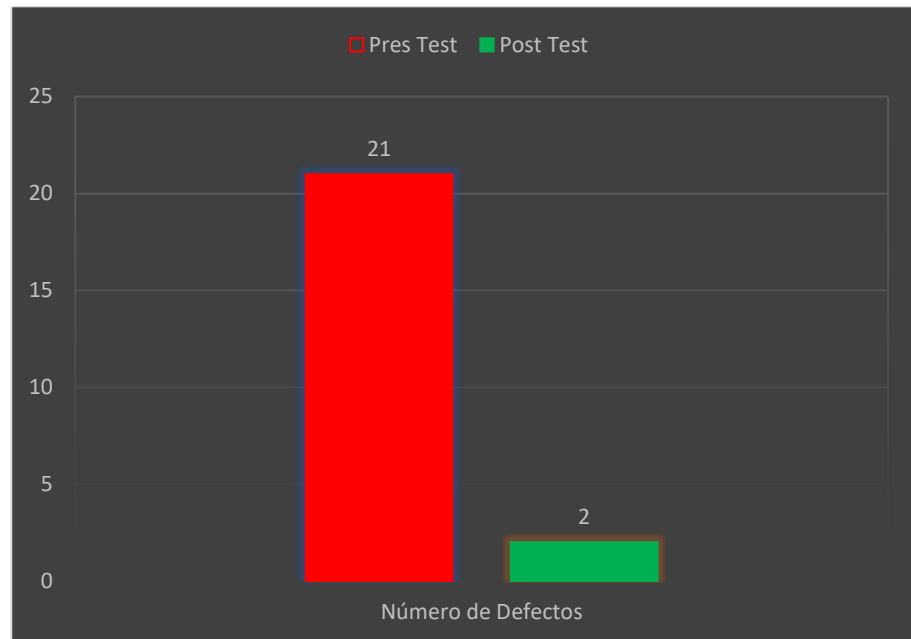


Figura 23, Cantidad promedio de defectos por proyecto

Fuente, elaboración propia.

3.1.2.2. Indicador Porcentaje de Clientes Satisfechos

Tabla 6

Indicador Satisfacción del Cliente

GRUPO	Fórmula	Resultados obtenidos de la encuesta	% Satisfacción
Pre Test	$P_{CS} = \frac{(TC - CI)}{TC} * 100$	$P_{CS} = \frac{(3 - 2)}{3} * 100$	33.33%
Post Test	$P_{CS} = \frac{(TC - CI)}{TC} * 100$	$P_{CS} = \frac{(3 - 0)}{3} * 100$	100.00%

Fuente, elaboración propia.

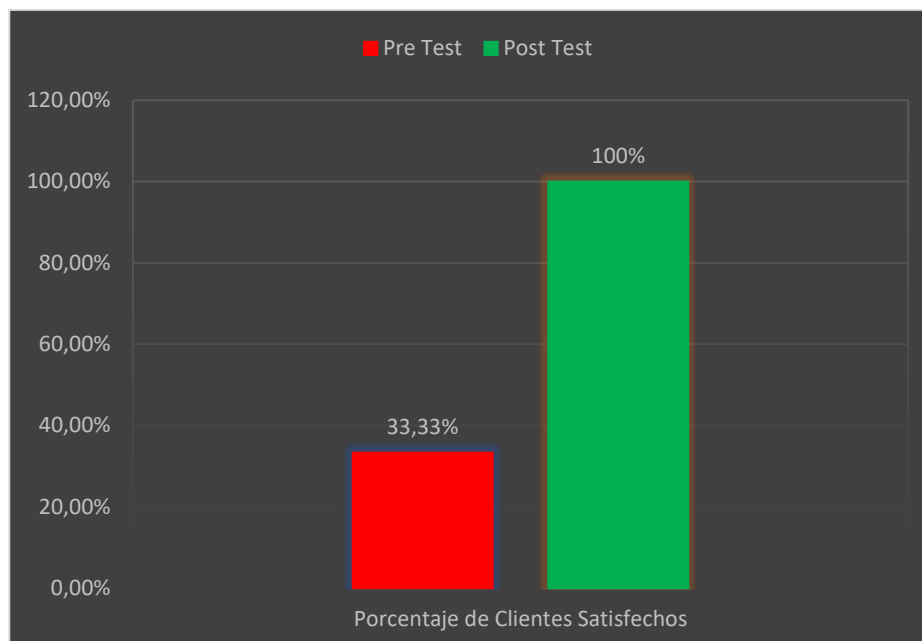


Figura 24, Indicador Satisfacción del Cliente

Fuente, Elaboración propia.

Se evidenció que, los proyectos que disponían de un mayor grado de satisfacción del cliente se encontraban situados en el grupo post-test con un máximo de 100.00 % en detrimento del grupo pre-test donde se obtuvo inicialmente un máximo de 33.33% de satisfacción del cliente, respecto a la entrega de servicios y/o productos software por parte de Sistema Inteligente ERP SAC.

3.2. Discusión de resultados.

Con respecto al objetivo general del presente informe de investigación, el cual fue desarrollar un modelo de procesos para la construcción de software en microempresas constructoras de software en el Perú, se pudo determinar que el modelo propuesto, en efecto, ayudó en la mejora de dicho proceso constructivo en microempresas, tal y como menciona Chevers (2017), quien aludió que, el mejoramiento de procesos de software busca consolidar una iniciativa que propicie la entrega de productos con altos estándares de calidad. Empero, a pesar de estos denodados esfuerzos, las tasas de uso, conocimientos y beneficios otorgados por estas inventivas para lograr el mejoramiento de los procesos

de la elaboración y construcción del software en las microempresas canadienses vienen siendo escasas.

Asimismo, se dio vital atención a la construcción de software, considerando que es un proceso de vital importancia y que es crítico a la vez, puesto que, si se es llevado de una manera inapropiada, puede causar innumerables pérdidas en la rentabilidad de las microempresas. Esto concuerda con lo mencionado por Laporte, Muñoz, & Gerançon (2017) quienes alegan que, fluyen incontables inconvenientes en los procesos de elaboración y construcción del software a nivel de calidad en los procesos mismos. Ante ello, es necesario la implementación y la usanza de estándares de nivel internacional, los mismos que permiten asegurar las entregas con productos o servicios de software con elevados estándares en cuanto a calidad.

Por otra parte, se pudo evidenciar que, con el modelo de procesos propuesto, en cuanto a la cantidad de defectos se obtuvo un máximo de 21 defectos obtenidos en el grupo pre test, mientras que en el grupo post test se obtuvo una reducción hasta llegar a los 2 defectos por grupo de proyecto, observándose una reducción determinante de 19 defectos por grupo de proyectos, lográndose eliminar defectos relacionados con la mala estimación de tiempos, la insuficiente administración de los riesgos, el escatimar en el control de calidad de los procesos constructivos, se diseñaba inadecuadamente, exceso de confianza hacia herramientas y tecnologías que no eran previo a su usanza y la añadidura de personal adicional a proyectos que se desfasaban en contraste con la línea de tiempo de entregas planificada; demostrando que, el modelo propuesto, basado en una combinación de la norma ISO/IEC 29110 y el modelo MoProSoft, conllevó a alcanzar una mejora en cuanto a las dimensiones de la variable calidad de los procesos para la construcción de software en la empresa Sistema Inteligente ERP SAC. Estos resultados concuerdan con lo mencionado por Coleman & O'Connor (2008) quienes alegaron que, las microempresas carecen de modelos de procesos desplegados para el desarrollo del

software, los cuales les sirvan de guía, por lo que prefieren en cambio dejar caer elementos de su modelo elegido o, desarrollar algo patentado que mejor se adapte a sus requerimientos específicos, lo cual se ha logrado en la presente investigación.

Finalmente, se pudo evidenciar que, con el modelo de procesos propuesto, en cuanto al porcentaje de usuarios satisfechos con el software desarrollado posterior al modelo de procesos se obtuvo un 100% de satisfacción, quedando demostrado que, dicho modelo basado en la norma ISO/IEC 29110 y el modelo MoProSoft, conllevó a lograr un mejoramiento en cuanto a la disminución de errores del software construido en la empresa Sistema Inteligente ERP SAC. Estos resultados concuerdan con lo mencionado por Clarke & O'Connor (2013) quienes mencionaron que, las PYME de software adoptan marcos de trabajo combinados para los procesos de elaboración y construcción del software, adoptando características de distintos enfoques de la construcción del software prescritos anteriormente, lo cual les permite lograr la satisfacción de sus requerimientos, a pesar de las diversas dificultades que esta puede tener en diferentes aspectos.

3.3. Aporte práctico.

3.3.1. Calidad para la construcción de software

Se desarrolló una investigación acerca de los modelos de calidad para la construcción de software en microempresas. Según Callejas, Alarcón & Álvarez (2017), toda organización debe de cumplir con un proceso que se convierta en soporte, de modo que se encuentre perfectamente documentado, y que se ampare en diversas praxis delimitadas según modelo, que sirva de soporte a la organizaciones con el propósito de obtener un mejoramiento continuo y que, “éstas logren ser mucho más calificadas y competentes, de modo que se pueda evaluar la calidad de sus procesos constructivos y así ofrecer productos o servicios de óptimos niveles” (Callejas et al., pág. 238).

Acerca de esto, existen diversos modelos que se encuentran estructurados de distintas maneras y por numerosos autores. Sin embargo, para el presente informe, se tuvo como prioridad la estructura delimitada por Callejas et al., (2017) quienes clasificaron a los modelos de calidad de acuerdo a tres factores, según el enfoque presentado en la Figura 25:

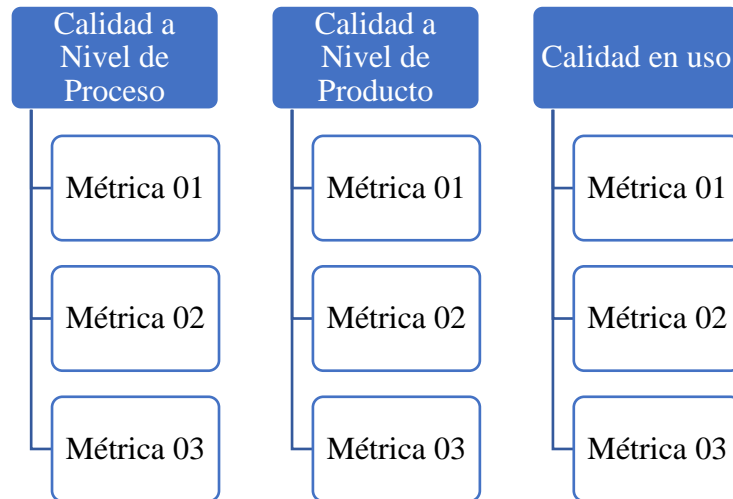


Figura 25, Estructura de la calidad de software.

Fuente, Callejas et al., (2017)

Los distintos modelos y estándares que buscan la gestión y aseguramiento en la calidad del software en el sector constructor de software, se encuentran definidos con el fin de lograr la contribución en la fabricación de servicios y/o productos con altos estándares e índices de calidad, los cuales se encuentren incluidos en el cronograma y los presupuestos fijados, de manera que se optimicen tanto los recursos, como los esfuerzos empleados (Muñoz, et al., 2019). Según la estructura mencionada anteriormente en la Figura 25 y, considerando el propósito de este informe, se buscaron solamente aquellos modelos destinados específicamente para la calidad de procesos y que tratan la construcción de software de una manera pertinente.

3.3.2. Modelos de Calidad para la construcción de software

Actualmente hay varios estándares y modelos de calidad de procesos para la construcción de software, tales como la ISO 29110, ISO 12207,

CMMI, MOPROSOFT y COMPETISOFT. Dichos modelos han sido identificados en base a la investigación bibliográfica que se ha realizado en los papers de bases de datos confiables, a las cuales se tuvo acceso gracias a los permisos brindados por la Universidad Señor de Sipán. Las bases de datos que se utilizaron como fuentes de información para realizar la búsqueda de los papers afines con modelos de procesos constructivos de software se encuentran en la Tabla 7.

Tabla 7

Fuentes de información.

N°	Base de datos	URL
1	EBSCO	https://www.ebsco.com/
2	IEEE Xplore	https://ieeexplore.ieee.org/
3	PROQUEST	https://www.proquest.com/
4	Science Direct	https://sciencedirect.com/
5	SCOPUS	https://www.scopus.com/

Fuente: elaboración propia

Asimismo, las cadenas de búsqueda se construyeron mediante las palabras claves, sirvieron para desplegar la respectiva búsqueda en las bases de datos bibliográficas indicadas y seleccionadas para la exploración de información. Dichas cadenas de búsquedas se encuentran delimitadas en la Tabla 8.

Tabla 8

Cadenas de búsqueda para cada base de datos.

N°	Base de datos	Cadena de búsqueda
1	EBSCO	("Process models" OR "Process methodology") AND ("Software development" OR "Software construction") AND ("Software improvement process" OR "Improve the development process") AND ("Microenterprises" OR "Very small entities")

2	IEEE Xplore	("Process models" OR "Process methodology") AND ("Software development" OR "Software construction") AND ("Software improvement process" OR "Improve the development process") AND ("Microenterprises" OR "Very small entities")
3	PROQUEST	("Process models" OR "Process methodology") AND ("Software development" OR "Software construction") AND ("Software improvement process" OR "Improve the development process") AND ("Microenterprises" OR "Very small entities")
4	Science Direct	("Process models" OR "Process methodology") AND ("Software development" OR "Software construction") AND ("Software improvement process" OR "Improve the development process") AND ("Microenterprises" OR "Very small entities")
5	SCOPUS	("Process models" OR "Process methodology") AND ("Software development" OR "Software construction") AND ("Software improvement process" OR "Improve the development process") AND ("Microenterprises" OR "Very small entities")

Fuente: elaboración propia

Paso siguiente se procedió a la búsqueda de información en las bases de datos mencionadas en la Tabla 7, lo que conllevó a encontrar quince (15) papers relacionadas con la temática desplegada en la presente investigación, los cuales se encuentran especificados a continuación en la Tabla 9.

Tabla 9
Papers relacionados con la usanza de modelos/estándares de construcción de software.

N°	Título Paper	Año	Autor(es)	País
1	Uso de modelos de calidad en las mypes productoras de software de Lima	2019	Amable & Millones	Perú
2	A Comparative Analysis of the Implementation of the Software Basic Profile of ISO/IEC 29110 in Thirteen Teams That Used Predictive Versus Adaptive Life Cycles	2019	Muñoz et al.	Escocia
3	Las PyME de desarrollo de software. Modelos de mejora de sus procesos en Latinoamérica	2019	Suárez & León	Venezuela
4	Implementación del Estándar ISO/IEC29110 en Centros de Desarrollo de Software de Universidades Mexicanas: Experiencia del Estado de Zacatecas	2018	Muñoz et al.	Colombia
5	Análisis de experiencias de mejora de procesos de desarrollo de software en PYMEs	2018	Coque et al.	Ecuador
6	Software process improvement: awareness, use, and benefits in Canadian software development firms	2018	Chevers	Brasil
7	Toward a simplified software process improvement framework for small software development organizations	2017	Chevers et al.	EEUU

8	Propuesta de modelo de mejora para mypes productoras de software	2017	Amable	Perú
9	The education of students about ISO/IEC 29110 software engineering standards and their implementations in very small entities	2017	Laporte et al.	Canadá
10	The Route to software process improvement in small and medium-sized enterprises	2017	Sánchez et al.	Suiza
11	Software Process Improvement in Very Small Organizations	2016	Larrucea et al.	EEUU
12	Quality assurance in the software development process using CMMI, TSP and PSP	2016	Espejo et al.	Portugal
13	An Evaluation of Software Development Practices among Small Firms in Developing Countries: A Test of a Simplified Software Process Improvement Model	2016	Chevers et al.	EEUU
14	Calidad de software en las mypes productoras de software en Lima	2015	Amable et al.	Perú
15	Propuesta de un proceso de investigación cuantitativa: Aplicación en la caracterización de las mypes productoras de software	2015	Amable	Perú

Fuente: elaboración propia

Luego de la búsqueda de papers relacionados al tema, se procedió a la identificación de los modelos en relación al tipo de organización al que se encontraban orientados, pudiéndose distinguir cinco (05) modelos que se usan para los procesos de construcción. Dichos modelos se encuentran listados en la Tabla 10.

Tabla 10

Modelos de procesos de construcción de software.

N°	Modelo	Proveedor	Problemática	Tipo Empresa
1	ISO 29910	ISO	Desarrollo de Software	Pequeñas organizaciones
2	ISO 12207	ISO	Desarrollo de Software	Pequeñas y medianas organizaciones
3	CMMI	ISACA	Desarrollo de Software	Empresas en General
4	MOPROSOFT	ISO	Desarrollo de Software	Pequeñas organizaciones
5	COMPETISOF	SEI	Desarrollo de Software	Pequeñas y medianas organizaciones

Fuente: elaboración propia

En un momento en el que la calidad del software es clave para la ventaja competitiva, el uso de los sistemas ISO/IEC y los estándares de ingeniería de software sigue limitado a algunos de los más populares, como ISO 9000. Las investigaciones que se analizaron en este informe mostraron que, las microempresas pueden tener dificultades para emplear Normas ISO/IEC para sus necesidades comerciales y para justificar la aplicabilidad de las normas a sus operaciones comerciales. En su gran mayoría estas micro empresas no se adaptan por no contar con recursos en cantidad de colaboradores, experiencia, costo y tiempos al establecer procesos del ciclo de la vida del software. A veces existe una desconexión entre la visión a corto plazo de la organización, que busca qué la mantendrá en el negocio durante otros seis meses más o menos, y los beneficios a largo plazo de mejorar gradualmente las formas en que la empresa puede administrar el desarrollo y mantenimiento de su software. Una razón principal citada por muchas organizaciones pequeñas de software para esta falta de adopción

de tales normas ISO fue la percepción de que han sido desarrolladas por y para grandes empresas multinacionales de software y no pensando en la pequeña organización. A continuación, se despliega un análisis de las normas anteriormente citadas:

Tabla 11

Tabla resumen de modelos de procesos de construcción de software.

Denominación	Proveedor	Nombre	Tipo de empresa	Status	Alcance
ISO/IEC 29110 - Ingeniería de Software	ISO	29110	Pequeñas organizaciones	Norma	Determina y mejora el trabajo en equipo para procesos de software.
ISO/IEC 12207 - Information Technology/Software Life Cycle Processes	ISO	12207	Pequeñas y medianas organizaciones	Estándar	Ciclo de vida del software.
Capability Maturity Model Integration	ISACA	CMMI	Empresas en General	Modelo	Modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de software
Modelo de Procesos para la Industria del Software	ISO	MOPROSOFT	Pequeñas organizaciones	Modelo	Referencia de procesos que contiene buenas prácticas para procesos de gestión de software.
Modelo de Mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica	SEI	COMPETISOFT	Pequeñas y medianas organizaciones	Modelo	Modelo de Mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica

Fuente, elaboración propia

Los estándares son fuentes de conocimiento codificado. Numerosos estudios han demostrado los beneficios de las normas, como una mayor productividad y calidad. Según lo declarado por Muñoz, Mejía, & Laporte (2018), “cuando una organización selecciona un estándar que se ajusta bien a su contexto y planifica la adopción cuidadosamente, es más probable que logre los beneficios anunciados del estándar” (Muñoz, Mejía, & Laporte, 2018, pág. 10). En la Tabla 12 se enumeran dos categorías de beneficios observados: beneficios observables en las actividades diarias del proyecto y beneficios para la pequeña organización como empresa:

Tabla 12

Beneficios Día-a-Día y beneficios comerciales.

Beneficios Día-a-Día	Beneficios Comerciales
a. Trabajo estandarizado y entregables consistentes	a. Mayor credibilidad para ofertar en licitaciones
b. entre proyectos	b. Acceso a mercados que requieran la certificación de un sistema de calidad acorde con las prácticas comerciales de la empresa
c. Evita reinventar la rueda para cada proyecto	c. Mejor reconocimiento de la calidad del trabajo realizado y de los productos desarrollados
d. El trabajo se realiza de forma sistemática y disciplinada.	d. Mayor confianza de los clientes y socios comerciales
e. camino	e. Un paso importante hacia un nivel de madurez del CMMI® (un nivel CMMI® es un requisito de algunos clientes)
f. Mejor calidad de entregables y productos	
g. Mejor gestión y seguimiento de proyectos	
h. Reducción de riesgos del proyecto	
i. Mejor comunicación dentro del equipo porque el	
j. la semántica de la comunicación está estandarizada	

Fuente: elaboración propia

Luego del análisis desplegado se optó, tal y como se mencionó en la muestra de estudio, por la Norma ISO/IEC 29110 y el modelo de procesos MoProSoft dado que, la estructura de sus marcos metodológicos, como se puede ver en la **¡Error! No se encuentra el origen de la referencia.**, permite la usanza por cualquier PYME constructora de software, además de la definición de modelos que faciliten la adopción e implementación de diferentes estándares, la fundamentación de una base holística de conocimientos que admitan a la compañía atesorar todo su capital intelectual que ha concebido por sí misma, sin la necesidad de acatar de una manera estricta del conocimiento individual, el aprovechamiento y la manipulación adecuada, ya sean de los colaboradores o de los recursos de naturaleza material que existen y, finalmente, la simplicidad que pudiese mostrar al momento de ser implantado en la compañía.

3.3.3. ISO/IEC 29110

Esta ISO es una norma que en un principio se desarrolló pensando en las microempresas desarrolladoras de software en el estado mexicano. Esta norma tiene como objetivo servir de soporte a las micro empresas al mejoramiento de su sistema o proceso constructivo de software, ayudándoles en la implantación progresiva de prácticas comprobadas para obtener sendos beneficios tales como el mejoramiento y aumento de la calidad de sus servicios o productos, reducir su tiempo de desarrollo, el porcentaje de re trabajo y desarrollar su producto, dentro del presupuesto y tiempo establecidos.

La mayoría de las normas internacionales en Ingeniería de Sistemas han sido elaboradas por y para grandes empresas. Las organizaciones pequeñas que intentarían cumplirlas fracasan la mayor parte del tiempo por falta de conocimiento o falta de recursos. A veces ni siquiera están convencidos de que la Ingeniería de Sistemas sea imprescindible para sus proyectos. Aunque ISO/IEC 29110 es una norma emergente, se han completado una serie de proyectos piloto en varios países utilizando algunos de los paquetes de implementación desarrollados. Esta norma ISO/IEC

29110 es ampliamente adaptable a pequeños contextos u organizaciones. Las pequeñas organizaciones son departamentos, entidades, organizaciones, compañías, empresas o proyectos que pueden constituirse con un máximo de 25 participantes. Para abordar el caso del caso de las PYMES, la reciente norma denominada “ISO/IEC 29110 Ingeniería de Sistemas e Software” tiene como propósito proporcionar un estándar ligero para las PYMES que desarrollan Software o Sistemas. Esta norma clasifica en cuatro perfiles a las pymes objetivo de acuerdo con el tamaño de los proyectos que se supone deben gestionar: “Perfil de entrada”, “Perfil básico”, “Perfil intermedio” y “Perfil avanzado”.

Tabla 13

Descripción de los perfiles ISO/IEC 29110.

Perfil	Objetivo
Entrada	Pymes de nueva creación (menos de tres años de funcionamiento) o pymes que trabajan en proyectos pequeños (menos de seis meses o personas).
Básico	Pymes desarrollando una aplicación única por un solo equipo de trabajo (contrato externo o interno).
Intermedio	Pymes involucradas en el desarrollo de más de un proyecto paralelamente y contando con no menos de un equipo de trabajo.
Avanzado	Pymes que pretenden sostenerse y crecer como un sistema competitivo independiente y / o negocio de desarrollo de software.

Fuente, elaboración propia

La norma se divulga públicamente, sin cargo de por medio, en el sitio web de ISO. La documentación práctica denominada “Deployment Package” o DP, compuesta por guías, ejemplos y plantillas, también está disponible de forma totalmente gratuita. Los PD brindan ayuda para ocho procesos, pero desafortunadamente solo para los dos primeros perfiles “Entrada” y “Básico”

(ver Tabla 13). Los DP para los otros dos perfiles "Intermedio" y "Avanzado" deberían estar disponibles en 2021. Este estándar de ingeniería de sistemas es bastante diferente del IEEE 1220, la ISO/IEC/IEEE 15288 y el ANSI/EIA 632 y el alcance es nuevamente diferente.

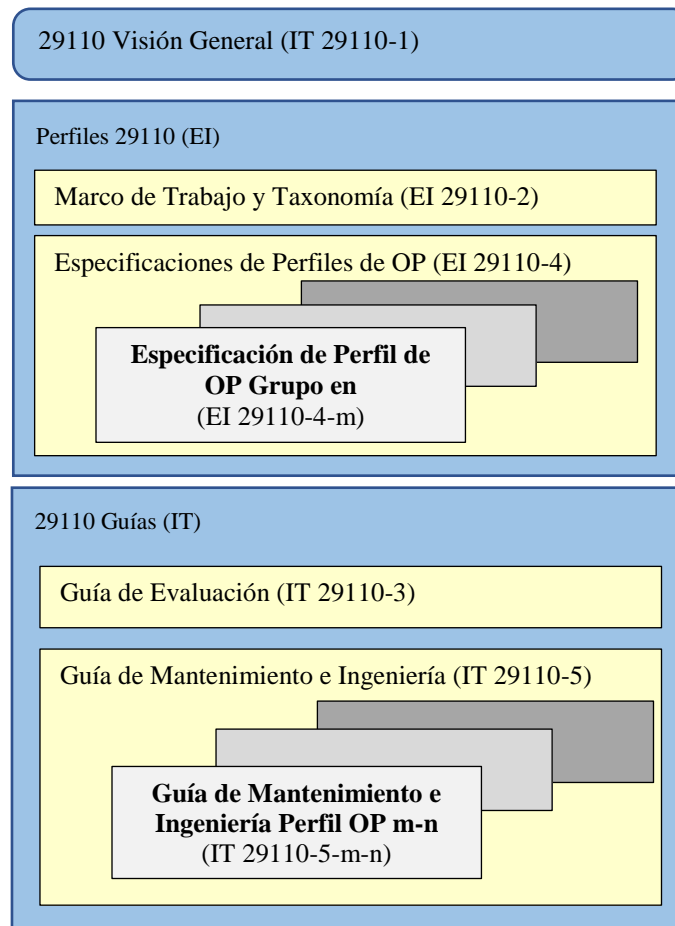


Figura 26, Modelo de procesos ISO/IEC 29110

Fuente, elaboración propia

A continuación, se describe en forma detallada cada uno de los documentos y perfiles que se integran en el marco de referencia de la ISO/IEC 29110.

ISO/IEC 29110-1, define los términos del negocio, para los documentos de los perfiles de las VSEs, creando los procesos, ciclo de vida, normalización, estandarización y documentos guía.

ISO/IEC 29110-2, define los términos comunes para los documentos del perfil VSEs, introduciendo la taxonomía o catálogo, menciona los elementos comunes de los perfiles estandarizados como estructura, conformidad y evaluación de los perfiles ISO/IEC 29110.

ISO/IEC TR29110-3, define las pautas de conformidad y las guías del proceso de la evaluación, necesarios para cumplir con el propósito de los perfiles definidos para la VSE. También incluye métodos y herramientas de evaluación para las personas que tienen relación directa con el proceso de evaluación.

El ISO/IEC TR29110-2, determina los elementos comunes de todos los perfiles como la estructura, conformidad, evaluación incluyendo la taxonomía de los perfiles.

ISO/IEC TR29110-4, Entrega la composición definitiva de un perfil, de los enlaces normativo de estándares usados en el perfil, y los enlaces de información o referencias a documentos de entrada.

Estos perfiles son subconjuntos de tareas de los estándares. Estos perfiles de las VSEs se aplican y están dirigidos a los autores, proveedores de guías, herramientas y otros materiales de soporte.

ISO/IEC 29110-5-m-n, provee guía de gestión e Implementación para el perfil VSEs descrito en ISO/IEC 29110-4-m.

Esta norma pretende dar solución a las pequeñas empresas, adaptadas a su tamaño y necesidad de negocio, para lo lograr competencia en el mercado, y aceptando los cambios tecnológicos que se producen.

La ISO 29110 es más práctica, está provista de muchas guías y ejemplos, es totalmente gratuita y, además, permite que las empresas estén certificadas. Existen numerosos documentos que ofrecen las razones

positivas para obtener la certificación. La mayoría de ellos distinguen los efectos positivos internos y los efectos positivos externos. Incluso si las empresas afirman que desean mejorar sus procesos internos, sus productos, en general, en primer lugar, quieren mejorar su imagen con respecto al mercado. Significa que la motivación puede tener diversos orígenes y también se puede clasificar en interna o externa. Un deseo de ISO 29110 es responder a esta necesidad básica: mejorar los procesos internos y externos.

Tabla 14

Deployment Package disponibles en ISO/IEC 29110

Id	Perfil de Entrada	Perfil Básico
1	Ingeniería de requisitos	Análisis de requerimientos
2	Gestión de interfaz	Construcción y pruebas unitarias
3	Gestión de la configuración	Control de versiones
4	Gestión de proyectos	Gestión de proyectos
5	Arquitectura funcional y física	Arquitectura funcional y física
6	Integración	Integración y pruebas
7	Despliegue de productos	Entrega del producto
8	Auto evaluación	Auto evaluación

Fuente, elaboración propia

Tal y como se mencionaba anteriormente, este estándar tiene un grupo de perfiles genéricos que proporciona una hoja de ruta de cuatro etapas para las pequeñas organizaciones que no construyen sistemas críticos o software crítico: perfiles de entrada, básico, intermedio y avanzado. Este estudio se basó en el perfil básico que representa las prácticas de desarrollo de un solo proyecto-aplicación por parte de un equipo único para dicho proceso de elaboración y construcción. El perfil básico tiene dos (02) procesos interconectados: Gestión de proyectos (PM) e Implementación de software (SI). Este estudio se centra en ambos procesos, tanto de PM, como de SI porque su objetivo es lograr productos y/o servicios de software que

satisfagan los requerimientos, intereses y expectativas de todos los usuarios potenciales, incluidas las cuestiones de seguridad.

3.3.3.1. Proceso de Gestión de Proyectos

Se conceptualiza como aspiración de este proceso nombrado Gestión de Proyectos (PM) al establecimiento y despliegue de tareas del proyecto de implantación del software de manera sistemática, lo que permite el cumplimiento de los objetivos del proyecto en términos de costes, tiempos y calidad esperados. La Figura 27 ilustra las 4 operaciones desplegadas dentro del proceso de Gestión de Proyectos.

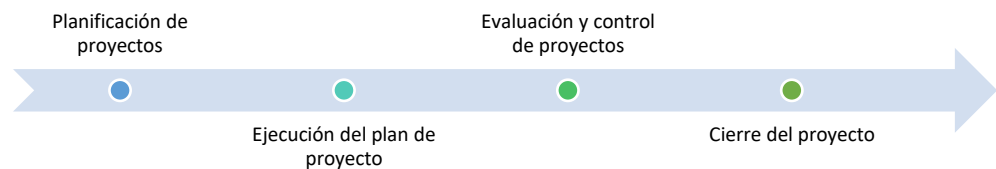


Figura 27, Proceso de Gestión de Proyectos.

Fuente, Elaboración propia

- a. **Planificación de proyectos:** El objetivo principal de este proceso es elaborar y anunciar planes prospectivos de proyectos que sean de naturaleza viable y efectiva. Esta planificación establece el alcance por parte de las actividades técnicas y la gestión del proyecto, logra la identificación de los resultados del proceso, las tareas del proyecto y los entregables, determina los calendarios de trabajo para la ejecución de las operaciones del proyecto, incluidos los recursos y los criterios de logro necesarios para realizar las tareas operativas del proyecto.
- b. **Ejecución del plan del proyecto:** Implementar las tareas de trabajo reales del proyecto teniendo en consideración el plan del proyecto. Viéndolo correctamente, una vez que el plan del proyecto se encuentra acordado y anunciado a todo el equipo y sus miembros, correspondería iniciar el trabajo de la elaboración y

producción del servicio y/o producto, el cual es el propósito real del proyecto.

- c. **Evaluación y control del proyecto:** La finalidad es establecer la actual situación del proyecto y garantizar que éste, se despliegue en concordancia con cronogramas y planificaciones, dentro de los presupuestos proyectados y que satisfaga los objetivos técnicos.
- d. **Cierre del proyecto:** Por lo general, implica conferir los entregables terminados a los usuarios o clientes, hacer entrega de documentos y legajos del proyecto a la compañía, desplegar la rescisión de contratos con los proveedores, libertar recursos del proyecto y desarrollar el anuncio de cierre de proyectos a la totalidad de partes inmiscuidas.

3.3.3.2. Proceso de Implementación de Software

El proceso de Implementación de Software (SI), ilustrado en la Figura 28, es lograr obtener una ejecución sistematizada correspondiente a las actividades de análisis, diseño, construcción, integración y prueba para productos de software nuevos o modificados de acuerdo con los requisitos especificados.

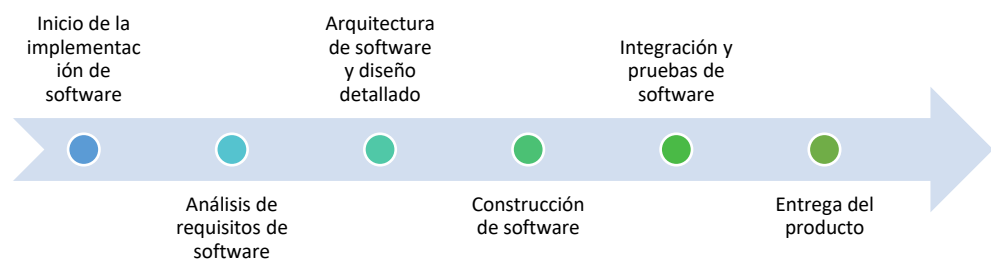


Figura 28, Proceso de Implementación de Software.

Fuente, Elaboración propia

- a. **El Inicio a la Implementación del Software** asegura que el Plan de Proyecto delimitado en la actividad de Planificación del Proyecto esté comprometido por el Equipo de Trabajo. Esta actividad

prepara al equipo para el resto de las actividades y reúne todos los instrumentos necesarios para realizar el proyecto. La limitación de las pequeñas empresas es el presupuesto limitado para la creación del entorno. En este punto, es importante estar familiarizado con para qué se puede utilizar cada herramienta o tecnología y cómo puede afectar el estado general de la seguridad del software.

- b. **El Análisis de Requisitos de Software** examina los requerimientos establecidos y pactados por parte del usuario-cliente y determina los requisitos del proyecto validado. Esta actividad estudia las necesidades y expectativas de los usuarios para definir el alcance del proyecto y distinguir las funcionalidades claves, incluidos los requisitos no funcionales, y la seguridad a menudo se considera como tal. Además, no vale la pena que las pequeñas empresas no puedan permitirse el lujo de tener expertos en seguridad y existen varios tipos de requisitos de seguridad que abordan los diversos principios de seguridad del software. No obstante, las necesidades de protección pueden obtenerse mediante varios métodos, entre los que se incluyen la lluvia de ideas, las encuestas, la descomposición de políticas, la clasificación de datos y el modelado de casos de uso y mal uso. El proceso de descomposición de políticas se compone de desglosar los requisitos de alto nivel en requisitos de seguridad de software de nivel más detallado. La clasificación de datos puede ayudar a garantizar que se asignen los niveles adecuados de controles de seguridad a los datos en función de sus niveles de sensibilidad. Finalmente, el modelado de casos de uso y mal uso, los diagramas de secuencia y los modelos sujeto-objeto se pueden utilizar para recopilar los requisitos de seguridad del software.
- c. **La Arquitectura de Software y Diseño Detallado** convierte los diversos requisitos del software en la arquitectura del software del sistema y el diseño detallado del software. Esta actividad es la

piedra angular del SDLC. No describir una arquitectura de diseño que incorpore todos los requisitos es una razón común para el fracaso del proyecto. Por lo tanto, las fallas semánticas o de lógica empresarial están relacionadas con problemas de diseño. La limitación de las pequeñas empresas es un equipo de desarrollo limitado. Cuando diseñan software, se tienen en cuenta las posibles amenazas y la seguridad, y deben tener en cuenta los principios de diseño seguro para garantizar la confidencialidad, disponibilidad e integridad. De hecho, el tiempo necesario para solucionar los problemas identificados es más corto cuando el software aún se encuentra en la fase de diseño. A continuación, se detallan algunas consideraciones que podrían ser útiles para hacer eso:

- Determinar los puntos de entrada y salida que un atacante podría usar para comprometer el activo de software o los datos que procesa. Tener en consideración los siguientes principios: privilegio mínimo, separación de funciones, defensa en capas, seguridad contra fallas, economía de mecanismos, mediación completa, diseño abierto, mecanismo menos común y aprovechamiento de los componentes existentes.
- Las consideraciones de diseño abordan los elementos de seguridad básicos de confidencialidad, integridad, disponibilidad, autenticación, autorización y auditoría.
- El proceso de diseño incluye la evaluación de la superficie de ataque, el modelado de amenazas, la identificación y priorización de controles y la documentación. Los modelos de amenazas son útiles para identificar y priorizar controles (salvaguardas) que se pueden diseñar, implementar (durante la fase de desarrollo) y desplegar, pero lleva tiempo.
- Se pueden aprovechar arquitecturas y tecnologías de software probadas para mejorar la seguridad del software. Por ejemplo, autenticación, gestión de identidad (IDM), gestión de

credenciales, control de flujo, auditoría / registro, prevención de pérdida de datos (DLP), virtualización y gestión de derechos digitales

- Revisión del diseño y la arquitectura del software desde una perspectiva de seguridad.

d. **La Construcción de Software** construye el código del software y los datos del Diseño de software. Esta actividad implica que los desarrolladores produzcan componentes utilizando un enfoque sistemático. Las limitaciones de las pequeñas empresas son un equipo de desarrollo limitado y poco tiempo para entregar. De hecho, escribir código seguro es un factor importante y crítico para garantizar la flexibilidad de los controles de seguridad del software. El mapeo realizado usando el método proporcionado se describe a continuación:

- Se deben tener en cuenta las ventajas de seguridad o la falta de metodología de desarrollo de software.
- Cree controles de protección de seguridad basados en vulnerabilidades de codificación comunes y una comprensión de cómo un atacante intentará explotar el software.
- Los procesos de construcción del software seguros incluyen control de versiones, análisis de código y revisión de código.
- Crear entorno y seguridad de herramientas. Los principales tipos de herramientas de compilación son compiladores, empaquetadores y empaquetadores, verbigracia, el instalador de Microsoft (MSI) y el administrador de paquetes de Red Hat (RPM).

e. **La integración y Pruebas de Software** garantizan que los elementos constituyentes de software integrados satisfagan los requisitos de software. Esta importante actividad comprende ejecutar un conjunto de pruebas y reconocer las complicaciones

que deben ser. Las limitaciones de las pequeñas empresas organizaciones son un equipo de desarrollo limitado. En este sentido, las pruebas de seguridad se pueden utilizar para determinar los medios y las oportunidades por las que se puede atacar el software. Estas pruebas son las siguientes:

- Pruebas de seguridad de caja blanca y caja negra, p. Ej. pruebas de fuzzing, escaneo y penetración: se utilizan para determinar las amenazas al software. Se basan en el conocimiento de cómo probar las vulnerabilidades comunes del software.
- Las pruebas relacionadas con problemas de seguridad del software son pruebas de validación de entrada, pruebas de fallas de inyección, pruebas de no repudio, pruebas de suplantación de identidad, pruebas de fallas, pruebas de validación criptográfica, pruebas de defensas de desbordamiento de búfer, pruebas de defensas de escalada de privilegios y pruebas de protección anti-reversión.
- El uso de herramientas aplicables a la situación específica. Algunas de las herramientas de seguridad comunes incluyen: herramientas de reconocimiento (recopilación de información), escáneres de vulnerabilidades, herramientas de huellas digitales, rastreadores/analizadores de protocolos, descifradores de contraseñas, herramientas de seguridad web, por ejemplo, escáneres, proxies, gestión de vulnerabilidades, herramientas de seguridad inalámbrica, herramientas de ingeniería inversa (Ensambladores y desensambladores, depuradores y descompiladores), analizadores de código fuente, herramientas de explotación de vulnerabilidades, sistemas operativos orientados a la seguridad y herramientas de prueba de privacidad.
- La reparación de defectos nunca debe realizarse directamente en el entorno de producción, y se deben utilizar

los principios de gestión de cambios adecuados para promover las correcciones desde los entornos de desarrollo y prueba a las pruebas de aceptación del usuario (UAT) y al entorno de producción.

f. **La Entrega del Producto** suministra al cliente el servicio y/o producto de software integrado. Esta actividad asegura que no habrá demoras para obtener la aceptación del producto para que el cliente complete el pago a la empresa. La limitación de las pequeñas organizaciones es el corto tiempo de entrega y la garantía de que el software no solo sea operacionalmente resistente a la piratería, sino que también cumpla con las regulaciones aplicables, es decir, tiene que existir un proceso formal en la aceptación del software que comprende la validación de los requisitos indispensables para la seguridad y la verificación de los controles de seguridad. En lo que sigue, consideraciones cruciales sobre la seguridad del software antes y después de la instalación.

- Los requisitos de seguridad deben validarse y los controles de seguridad deben verificarse mediante pruebas de seguridad internas o independientes de terceros. El software no debe implementarse y/o liberarse hasta que se haya certificado y acreditado que el riesgo residual está en el nivel apropiado.
- El endurecimiento del software implica: eliminar los ganchos de mantenimiento antes de la implementación, eliminar el código de depuración y las marcas en el código, cambiar la forma de escribir el código para que no contenga información confidencial, es decir, comentarios innecesarios, código colgante o información confidencial de comentarios en el código.

3.3.4. MoProSoft

El Modelo de Procesos para la Industria de Software (MoProSoft) en el Estado de México, tuvo su inicio en el 2003 con el fin de promover la estandarización en todas las organizaciones mexicanas constructoras y fabricantes de software mediante la presentación de excelentes prácticas de manufactura en lo que concierne a la gestión y administración de la ingeniería del software en sus procesos y, el cual ha sido ascendido al nivel de norma en el año 2005, por lo que desde ese preciso momento se convirtió en la norma mexicana NMXI-059/02-NYCE-2005. MoProSoft busca suministrar un modelo que se encuentre fundamentado en la adopción de excelentes prácticas de manufactura de carácter internacional, de modo que pueda lograr estandarizar aquellos procesos de elaboración y construcción del software en el estado mexicano, teniendo en consideración que esta industria del software en tierras aztecas se encuentra conformada en su gran mayoría pequeñas y medianas compañías. Entre los beneficios del modelo MoProSoft tenemos:

- a. Mejora la calidad del software en la organización que utiliza el modelo.
- b. Mejora la calidad de los servicios, alcanzando competitividad de nivel internacional.
- c. Alinea los objetivos estratégicos en todas las áreas de la organización.
- d. Impulsa el reconocimiento del nivel de madurez de procesos en las organizaciones.
- e. Proporciona a las organizaciones, el acceso de prácticas de ingeniería de software de calidad internacional.
- f. Permite a las empresas desarrolladoras de software que adoptan el modelo, ser referenciadas por clientes, autoridades y competidores.

El modelo MoProSoft está dirigido al desarrollo y mantenimiento de software. Permite a las organizaciones que lo desarrollan, implementar de mejor manera otros modelos como ISO/IEC 29110, ISO/IEC 15504, CMMI-

Dev, ISO9001:2008 así como ISO/IEC 12207. Es un modelo que pretende ser sencillo de entender e implementar. El documento del modelo MoProSoft contiene menos de 200 páginas, mucho más corto que los demás modelos, porque se hace muy práctico de aplicar. Fue creado para las organizaciones mexicanas dedicadas al desarrollo y mantenimiento de software.

MoProSoft busca contribuir efectivamente al cumplimiento de los objetivos de una organización, a través de la mejora de procesos, dejando de ser solo un marco de referencia. El modelo propone que la capacitación, adopción y evaluación tengan un bajo costo para las organizaciones. MoProSoft considera que el conjunto de procesos y practicas están relacionadas entre sí, y se llevan a cabo mediante roles, en el cual emplean recursos y contando con insumos, elaboran un satisfactor de negocio para el cliente (internos y externos). Cuando el modelo se refiere a un satisfactor de negocios, está señalándose a un producto o salida que debe entregarse o que es requerido por algún cliente, el cual puede ser externo o un área interna de la empresa. Y cuando se menciona a un insumo, se está refiriendo a una entrada que requiere un proceso, el cual puede ser un documento, un acta, un proceso anterior. Este modelo busca ser un soporte para las compañías en la integración del mejoramiento continuo, en la evaluación de la eficacia de la misma y, además, en la estandarización de sus prácticas mediante sus tres categorías a los que hace referencia:

- a. Gerencia de Alta Dirección: Aborda aquellas prácticas desplegadas por la alta dirección concernientes con la gestión empresarial. Suministra directrices para el funcionamiento del proceso contenidas en la categoría de procesos de gestión.
- b. Gerencia de Gestión: Aborda la gestión de procesos, prácticas de proyectos y recursos en concordancia con los lineamientos brindados por la Alta Dirección y suministra los elementos para los procesos de categoría de operación. Se constituye por tres (03) procesos: procesos de gestión de los procesos, procesos

de gestión de los proyectos y procesos de gestión de los recursos.

c. Gerencia de Operación: Aborda la gestión de procesos específicos de un proyecto y las prácticas de construcción y mantenimiento en concordancia con los elementos suministrados por la Dirección. Se constituye por dos (02)

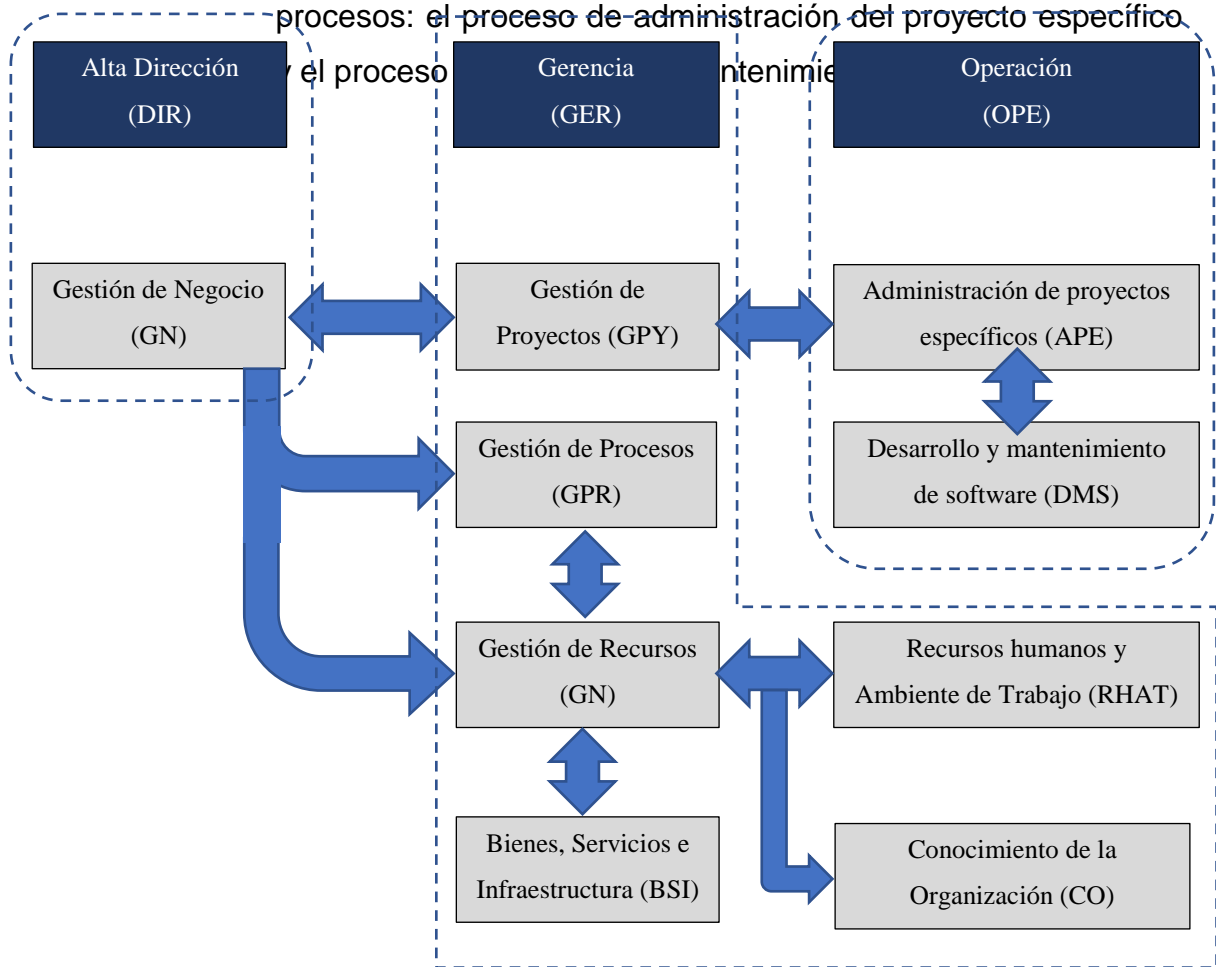


Figura 29, Categorías y Procesos de MoProSoft.

Fuente, Oktaba (2005).

MoProSoft define los Roles como una responsabilidad, la cual es tomada por un personal de la organización, pudiendo desempeñar más de un rol. Esta característica del modelo se debe a que MoProSoft se orienta hacia las pequeñas organizaciones, las cuales cuentan con un número

limitado de colaboradores, de manera que el modelo no pretende obligar a una organización a elevar sus costos en recursos humanos, si no por el contrario, se busca respetar la realidad y limitaciones.

MoProSoft señala los roles responsables para ejecutar las actividades de cada proceso. Los roles son asignados en función de las habilidades y capacitación de cada colaborador. Los roles están agrupados de acuerdo a las características y funciones a desempeñar, teniendo a los roles del Grupo directivo, los responsables de proceso y otros roles involucrados, además se tiene en cuenta al Cliente y al Usuario como roles externos. En la siguiente tabla se observa una rápida descripción de los roles mencionados.

Tabla 15

Descripción de Roles en MoProSoft

Rol	Descripción
Cliente	Es la persona u organización que solicita un producto software y se encarga del costo del proyecto
Usuario	Es la persona o grupo que usará el producto.
Grupo Directivo	Son los encargados de dirigir la organización
Responsable de Proceso	Este rol es desempeñado por el colaborador que realiza una determinada actividad y está a cargo de su cumplimiento
Involucrado	Son los colaboradores que desempeñan determinadas tareas de acuerdo a sus conocimientos (Analista, Programador, Revisor)

Fuente, elaboración propia

El enfoque de tres capas refleja la estructura de la mayoría de los contextos u organizaciones constructores de software en México. En consecuencia, a fines de 2010 más de 300 organizaciones en México fueron certificadas bajo NMX-I-059-NYCE-2005. Asimismo, esta norma mexicana se constituye de cuatro (04) partes:

- a. Parte 01 Definición de conceptos y productos
- b. Parte 02 Requisitos del proceso (MoProSoft)
- c. Parte 03 Directrices de implementación del proceso
- d. Parte 04 Principios de evaluación (EvalProSoft)

El trabajo de normalización fue estrictamente coordinado por la Asociación de Normalización y Certificación de Electrónica (NYCE). Los procesos de MoProSoft establecen seis niveles de madurez:

- a. Incompleto: el proceso no está completo o no logró su propósito.
- b. Realizado: el proceso se implementa y sus objetivos se logran.
- c. Gestionado: el proceso se gestiona y los productos resultantes se establecen, controlan y mantienen.
- d. Establecido: el proceso gestionado, basado en un proceso estándar, es usado.
- e. Predecible: el proceso instaurado logra ejecutarse dentro de límites definidos para lograr los resultados de su proceso.
- f. Optimizado: el proceso predecible se mejora de forma continua para lograr los objetivos comerciales presentes y futuros.

MoProSoft también contempla una sección en la que se describen los tipos de Productos que se proporcionan al implementarse el modelo. Así tenemos al Producto de software (generado por el Proceso de desarrollo y mantenimiento de software), Configuración de software (conjunto de productos software), los Planes, Reportes, Registros, Lecciones aprendidas, y otros generados en algún proceso como pueden ser los Contratos, Propuestas Tecnológicas, Documentación de Procesos, entre otros. En el Capítulo 6, MoProSoft determina el tipo de Uso del modelo de procesos que debe hacer una organización de acuerdo a su situación, de manera que una organización que no cuente con procesos establecidos, podrá adoptar el modelo, generando una instancia de cada proceso de MoProSoft. En una entidad que se esté aplicando procesos establecidos, se tiene que realizar una comparación de los procesos existentes con los de MoProSoft para

identificar similitudes y diferencias, luego de lo cual, se procede a hacer los ajustes que permitan adoptar el modelo de forma completa. Por último, se considera la Implantación y mejora continua, para que una organización vaya logrando sus objetivos al aplicar las sugerencias necesarias en sus procesos.

Modelo de procesos MoProSoft - Esquema del documento del modelo

Capítulo 1. Prologo
 Capítulo 2. Introducción
 Capítulo 3. Definiciones
 Capítulo 4. Patrón de Procesos
 Capítulo 5. Estructura
 Capítulo 6. Uso del modelo
 Capítulo 7. Categoría de Alta Dirección (DIR)
 Capítulo 8. Categoría de Gerencia (GER)
 Capítulo 9. Categoría de Operación (OPE)

Capítulo 4. Patrón de Procesos

Define el formato en que se describirán los nueve procesos

Definición general del proceso

Nombre
 Categoría
 Propósito
 Descripción general de sus actividades
 Objetivos
 Indicadores
 Metas cuantitativas
 Responsabilidad y autoridad
 Subprocesos
 Procesos relacionados
 Entradas
 Salidas
 Productos internos
 Referencias bibliográficas
Prácticas
 Roles involucrados
 Capacitación requerida
 Actividades detalladas con su abreviatura
 Abreviatura del rol que la desempeña
 Diagrama de flujo de trabajo del proceso
 Verificaciones y validaciones requeridas
 Productos
 Recursos de infraestructura necesarios
 Mediciones de procesos
 Prácticas de capacitación
 Situaciones excepcionales
 Uso de lecciones aprendidas

Guías de Ajuste

Se definen cambios a los procesos sin afectar el propósito de los mismos

Capítulo 3. Definiciones

Es una especie de diccionario de cada concepto utilizado en el documento

Capítulo 5. Estructura

Define las categorías, procesos, roles y productos del modelo.

Capítulo 6. Uso del modelo

Describe la forma de adoptar el modelo según el tipo de organización donde se implemente.

Capítulo 7. Categoría de Alta Dirección (DIR)

Procesos
 Gestión de Negocio

Capítulo 8. Categoría de Gerencia (GER)

Procesos
 Gestión de Procesos
 Gestión de Proyectos
 Gestión de Recursos
 Recursos Humanos y Ambiente de Trabajo
 Bienes, Servicios e Infraestructura
 Conocimiento de la Organización

Capítulo 9. Categoría de Operación (OPE)

Procesos
 Administración de Proyectos Específicos
 Desarrollo y Mantenimiento de Software

3.3.5. Comparación de ISO/IEC 29110 y MoProSoft

Las tareas y productos de trabajo de cada proceso de ISO/IEC 29110 se enfrentaron a las tareas y productos de trabajo del proceso equivalente en MoProSoft, como se revela en la Tabla 16.

Tabla 16

Equivalencia entre procesos involucrados en el mapeo.

ISO/IEC 29110		MoProSoft
Gestión de Proyectos	↔	Gestión de Proyectos Específicos
Implementación de Software	↔	Desarrollo y Mantenimiento de Software

Fuente, elaboración propia

Por otra parte, al referirnos a un producto de trabajo, lo hacemos realmente hacia un artefacto generado al ejecutar una tarea o un conjunto de tareas. Ambos estándares incluyen una sección que enumera cada producto de trabajo y sus características. Las secciones de MoProSoft son: Entradas, Salidas y Productos internos, mientras que la sección de ISO/IEC29110-5-1-2 es: 9. Descripción del producto.

Asimismo, un rol es un conjunto de competencias, habilidades, conocimientos y destrezas requeridas para realizar una tarea en una situación específica. Al final, para completar el mapeo, se analizaron los roles requeridos en ambos estándares. La sección de MoProSoft que define los roles es: Roles involucrados y entrenamiento, mientras que la sección ISO/IEC 29110-5-1-2 es: 8. Roles.

A continuación, se muestra las diferencias entre el perfil básico de ISO/IEC 29110 y MoProSoft.

- i. El perfil básico de ISO/IEC 29110 suministra un número menor de prácticas a implementar. Proporciona la cantidad mínima de actividades que se deben esperar para la construcción de

software por parte de equipos pequeños. Por tanto, su implementación en equipos pequeños podría presentar menos resistencia al cambio que MoProSoft.

- ii. El perfil básico de ISO / IEC 29110 se centra solo en las prácticas para realizar un proyecto a la vez.
- iii. El perfil básico de ISO/IEC 29110 proporciona prácticas relacionadas con el aseguramiento de la calidad del proceso de implementación del software, como las prácticas de validación y verificación.
- iv. El perfil básico de ISO/IEC 29110 destaca las actualizaciones del registro de trazabilidad de cada actividad del proceso de implementación del software.
- v. El perfil Básico de ISO/IEC 29110 asegura que la interacción de los diferentes miembros del equipo a través de prácticas tituladas “Asignar Tareas a los miembros del Equipo de Trabajo de acuerdo con su rol” a lo largo del proceso de implementación del software.
- vi. MoProSoft no solo circunscribe actividades afines con la gestión e implementación de proyectos, también envuelve operaciones afines con la gestión organizacional y la gestión empresarial.
- vii. MoProSoft se puede utilizar para realizar más de un proyecto a la vez.
- viii. MoProSoft asegura que la estimación de tiempos y presupuesto pactados se ajusten a la inversión que el cliente considera destinar para la adquisición del proyecto recomendando el desarrollo del plan de ventas.
- ix. MoProSoft asegura el establecimiento de los canales de comunicación con el cliente para realizar validaciones, así como su conocimiento de cambios.
- x. Por su alcance, además del desarrollo del plan del proyecto, MoProSoft se enfoca en el desarrollo de planes completos para: ventas, adquisición y capacitación, comunicación, desarrollo y prueba.

A continuación, se muestran las similitudes entre el perfil básico de ISO/IEC 29110 y MoProSoft:

- i. Ambos estándares buscan una estandarización de la gestión y control del desarrollo de software a través de prácticas probadas en ingeniería de software y gestión de proyectos.
- ii. Ambos estándares proporcionan actividades relacionadas con la planificación, el seguimiento y el control, la gestión y los requisitos de desarrollo, la verificación, la validación, la gestión de la configuración y la gestión de riesgos dentro de la gestión de proyectos y el desarrollo de software.
- iii. Ambos estándares se definen utilizando elementos de proceso comunes y brindan información sobre procesos, roles, actividades y productos de trabajo (es decir, documentos a producir como los requisitos).
- iv. Ambos estándares siguen un proceso de gestión que brinda transparencia a las acciones correctivas utilizadas para corregir los defectos detectados durante el proyecto.
- v. Ambos estándares proporcionan un proceso de implementación de software sistemático que busca satisfacer las necesidades del cliente y asegurar la calidad de los productos desarrollados.

3.3.6. Caso de Estudio: Sistema Inteligente ERP SAC

Para el presente proyecto, se hizo uso de un caso de estudio que fue la empresa Sistema Inteligente ERP SAC. Sistema Inteligente ERP SAC, es una compañía limeña de asesoría, consultoría y servicios en tecnología de información con 10 años de experiencias en la implementación de productos y herramientas informáticas originales e innovadoras a los servicios de muchas empresas y organizaciones en general. Durante este tiempo se ha especializado en desarrollar aplicaciones informáticas y dar soluciones, con enfoque en el mejoramiento continuo de los procesos que, como organización tiene como tal, desarrollando herramientas de software que

permite integrar todas las áreas de la organización y garantiza la seguridad informática, automatizando la totalidad del control documentario de sistemas de manera integral.



Figura 30, Logo Sistema Inteligente ERP SAC.

Fuente, Elaboración propia.

Sistema Inteligente ERP SAC se decanta siempre por el aprovechamiento de las tecnologías de primer nivel, por el buen desarrollo de sus productos y servicios y, sobre todo, por la garantía en la calidad con la que cuentan sus productos finales. Brinda servicios los cuales cumplen con los requerimientos y con las expectativas establecidas por su clientela, de la mano con un compromiso que garantiza la responsabilidad social corporativa y, sobre todo, la conservación del medio ambiente. Por su naturaleza, Sistema Inteligente ERP SAC ofrece productos y servicios los cuales se enfocan en mejorar los procesos y operaciones que llevan a cabo sus clientes en el día a día. Entre estos servicios y producto tenemos:

- a. Consultoría Informática
- b. Desarrollo de Software
- c. Ingeniería de Redes y Sistemas
- d. Venta de Equipos de Cómputo, componentes, Suministros

Para garantizar una gestión con resultados exitosos, Sistema Inteligente ERP SAC ha formado un equipo interdisciplinario de profesionales para poder analizar tanto los procesos productivos como los administrativos, y así brindar un servicio de consultoría y desarrollo de software que le permite ofrecer a sus clientes la mejor solución de negocio adecuada a sus necesidades, así como a sus posibilidades.

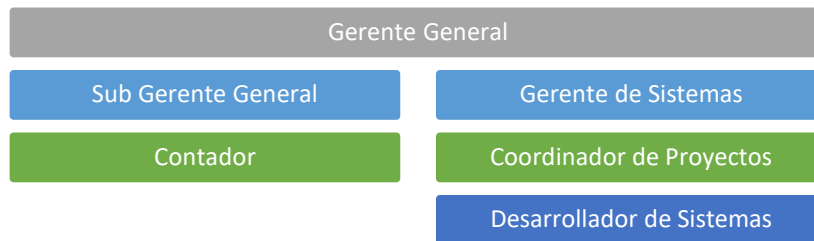
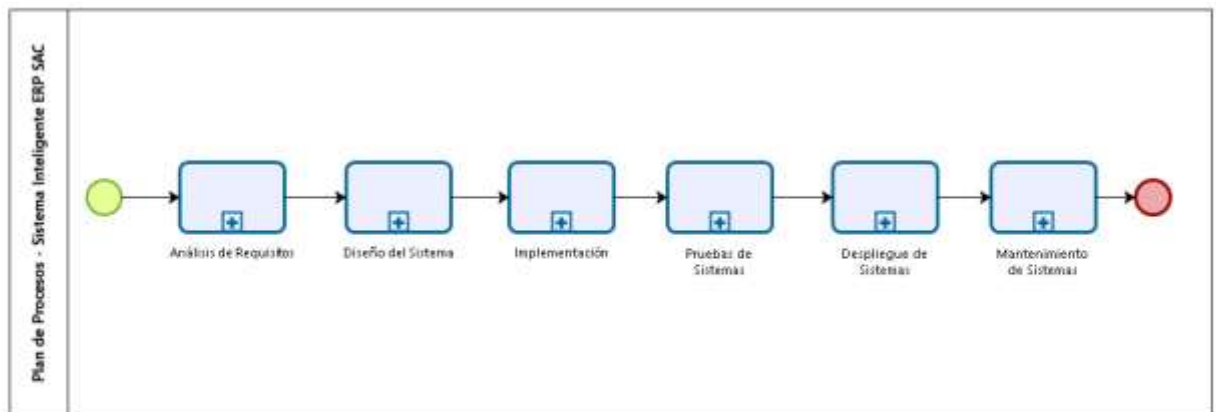


Figura 31, Organigrama empresarial Sistema Inteligente ERP SAC.

Fuente, Elaboración propia

A continuación, se muestra el Plan de Procesos que presenta Sistema Inteligente ERP SAC actualmente para la construcción de software:



Presented by
bizagi
MAGISTER

Figura 32, Plan de Procesos Sistema Inteligente ERP SAC.

Fuente, Elaboración propia

3.3.7. Modelo de Proceso Propuesto

El modelo propuesto, que se despliega en la Figura 33, es el resultado de la revisión de la literatura, caracterización de los modelos seleccionado en la muestra y el análisis de los mismos. El modelo propuesto tenía como propósito desarrollar e implementar en microempresas peruanas desarrolladoras de software un modelo adherido a la Norma ISO/IEC 29110 y el Modelo MoProSoft, contemplando las fases con las que cuentan y adaptadas al escenario peruano.

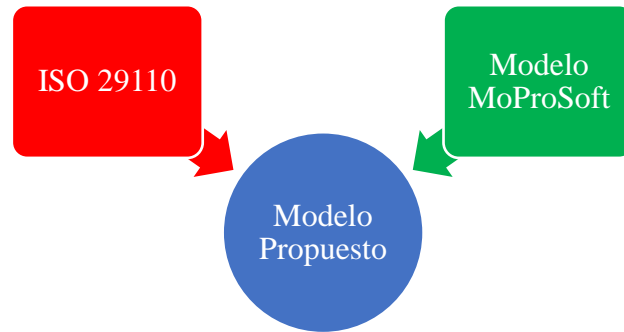


Figura 33, Modelo de construcción de software propuesto.

Fuente, Elaboración propia

Para cumplir con el propósito de la investigación, se definió un método específico para la construcción del modelo, la cual se encuentra representada en la Figura 34. Se definen tres actividades secuenciales a saber:

- a. En primer lugar, se describió el modelo de perfil básico ISO/IEC 29110.
- b. En segundo lugar, se mapeó del perfil básico ISO/IEC 29110 al lenguaje ArchiMate.
- c. En tercer lugar, el Modelo ArchiMate de perfil básico ISO/IEC 29110 se desarrolló como una instancia prototípica del mapeo propuesto.

Luego, el modelo se evaluó mediante juicio a expertos. Las actividades de investigación se describen en detalle en las siguientes secciones.



Figura 34, Actividades para el desarrollo del Modelo.

Fuente, Elaboración propia

Aunque ArchiMate se describe con precisión por su modelado, por otro lado, la norma ISO/IEC 29110 y el Modelo MoProSoft no incluyen ningún modelo en su contenido. Kabaale, Wen, Wang, & Rout (2017) presentaron un modelo basado en axiomas para la formalización de procesos de software y lo ilustraron modelando los procesos ISO/IEC 29110. Sin embargo, para la presente investigación, se requirió desarrollar un modelo específico para microempresas peruanas y basado en el estándar ISO/IEC 29110 y en el Modelo MoProSoft. En vista del hecho de que la norma ISO/IEC 29110 tiene muchas partes. Por lo tanto, el mapeo se limitó al Perfil Básico más utilizado para la ingeniería de software para demostrar la prueba de concepto. Dicho modelo desarrollado para el perfil básico ISO/IEC 29110 se presenta en la Figura 35.



Figura 35, Procesos del Perfil Básico ISO/IEC 29110.

Fuente, Elaboración propia

Como se puede apreciar, el perfil básico tiene dos procesos, es decir, procesos de gestión de proyectos e implementación de software. Esto concuerda con lo revelado por Oktaba (2005) referente al Modelo de Procesos para la Industria de Software - MoProSoft, quien mencionó que, el

Enunciado de Trabajo de los productos de trabajo clave se presenta como una entrada y la Configuración del software como una salida.



Figura 36, Procesos del Perfil Básico MoProSoft.

Fuente, Elaboración propia

Por lo mostrado previamente, tanto para la norma ISO/IEC 29110 y el Modelo MoProSoft, se consideró desplegar el modelo propuesto en dos procesos básicos: a. Proceso de Gestión de Proyectos y, b. Proceso de Implementación de software.

El Proceso de Gestión de Proyectos propuesto para este Modelo de construcción de software tuvo en su composición cuatro (04) actividades a las que se les denominó:

- a. PGP01 Planificación del Proyecto
- b. PGP02 Ejecución del Plan del Proyecto
- c. PGP03 Evaluación y Control del Proyecto
- d. PGP04 Cierre del Proyecto.

Todos los productos de trabajo utilizados por el proceso de gestión de proyectos se muestran agrupados en los grupos de productos de entrada, internos, de entrada y salida y de salida. Este proceso de Gestión de Proyectos para el Modelo de Construcción de Software propuesto se encuentra modelado en la Figura 37. Asimismo, los Objetivos del Proceso de Gestión de Proyectos propuesto para este Modelo de Construcción de

Software tiene en su composición siete (07) objetivos a saber y que se encuentran modelados en la Figura 38.

Las finalidades o metas del proceso de Gestión de Proyectos se citan a continuación:

Tabla 17

Objetivos del proceso de Gestión de Proyectos del Perfil Básico ISO/IEC 29110.

Objetivo	Descripción
PGP.O1	El Plan de Proyecto para el despliegue del proyecto se constituye en concordancia con la Declaración de Trabajo y es examinado y, posteriormente admitido por el usuario-cliente. Los recursos y las tareas fundamentales para lograr el cumplimiento del trabajo están estimados y correctamente dimensionados.
PGP.O2	El monitoreo del avance del proyecto se constata con la planificación del proyecto y se anota en el registro de estado de progreso.
PGP.O3	Los requerimientos de cambio se atienden por intermedio del análisis y la recepción. Estos cambios por parte de los requisitos de software se calculan en función del costo, el calendario y el impacto técnico.
PGP.O4	Se efectúan reuniones de monitoreo supervisión con el usuario-cliente y el equipo de trabajo. Los acuerdos se registran y se rastrean.
PGP.O5	Mientras se ejecuta el proyecto, vamos identificando los riesgos que estos puedan presentarse.
PGP.O6	Se cuenta con versiones de software, para obtener una mejor estrategia de control. Los componentes de la configuración del software se identifican, definen y establecen un eje base. Las alteraciones y liberaciones de la documentación se inspeccionan y sitúan a la orden del cliente y equipo de trabajo,

incluido el almacenamiento, manipulación y otorgamiento de los artículos.

PGP.O7 Para obtener un software a la medida y exigencia del cliente, la garantía de calidad del software se efectúa para garantizar un buen cumplimiento de los productos y procesos de trabajo, según el plan del proyecto y la especificación de requisitos.

Fuente, elaboración propia.

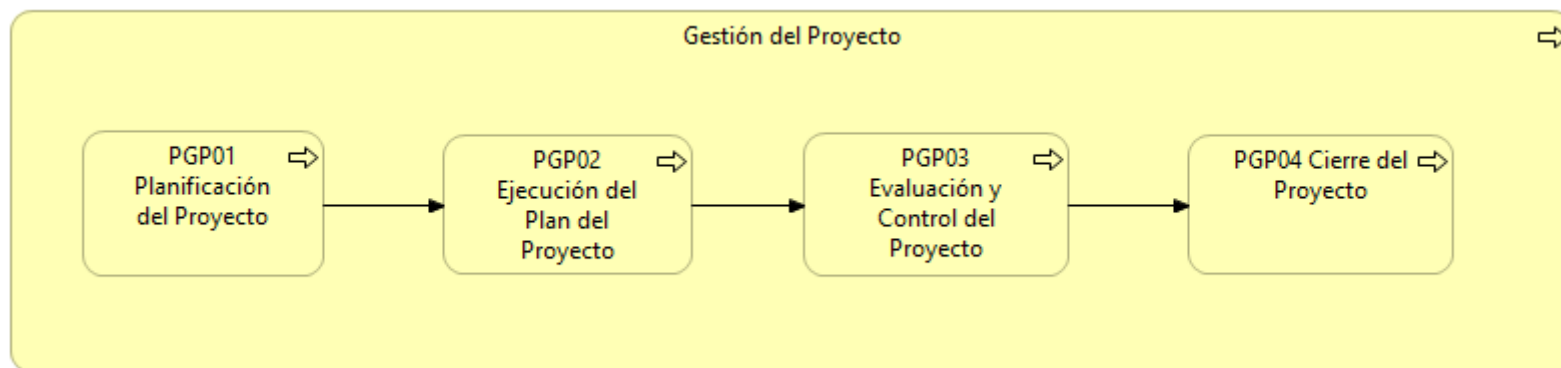


Figura 37, Vista del Proceso de Gestión de Proyectos.

Fuente, Elaboración propia

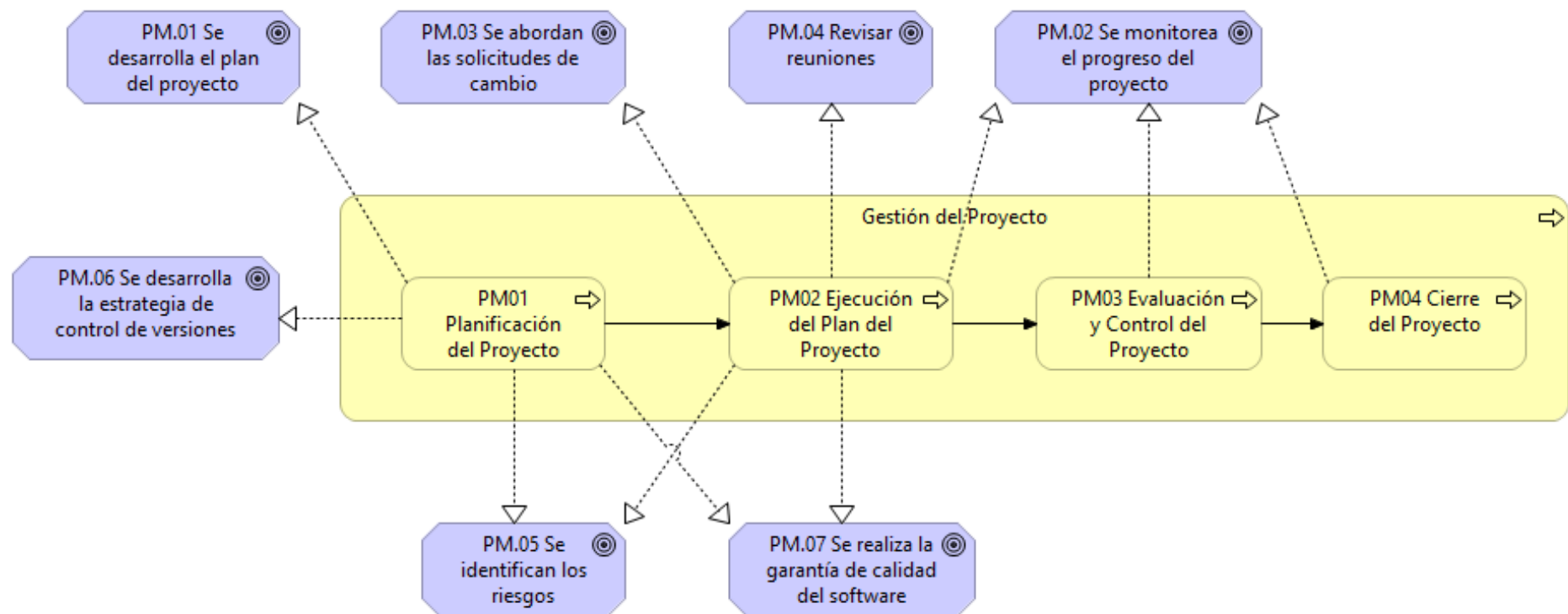


Figura 38, Objetivos de la vista del Proceso de Gestión de Proyectos.

Fuente, Elaboración propia

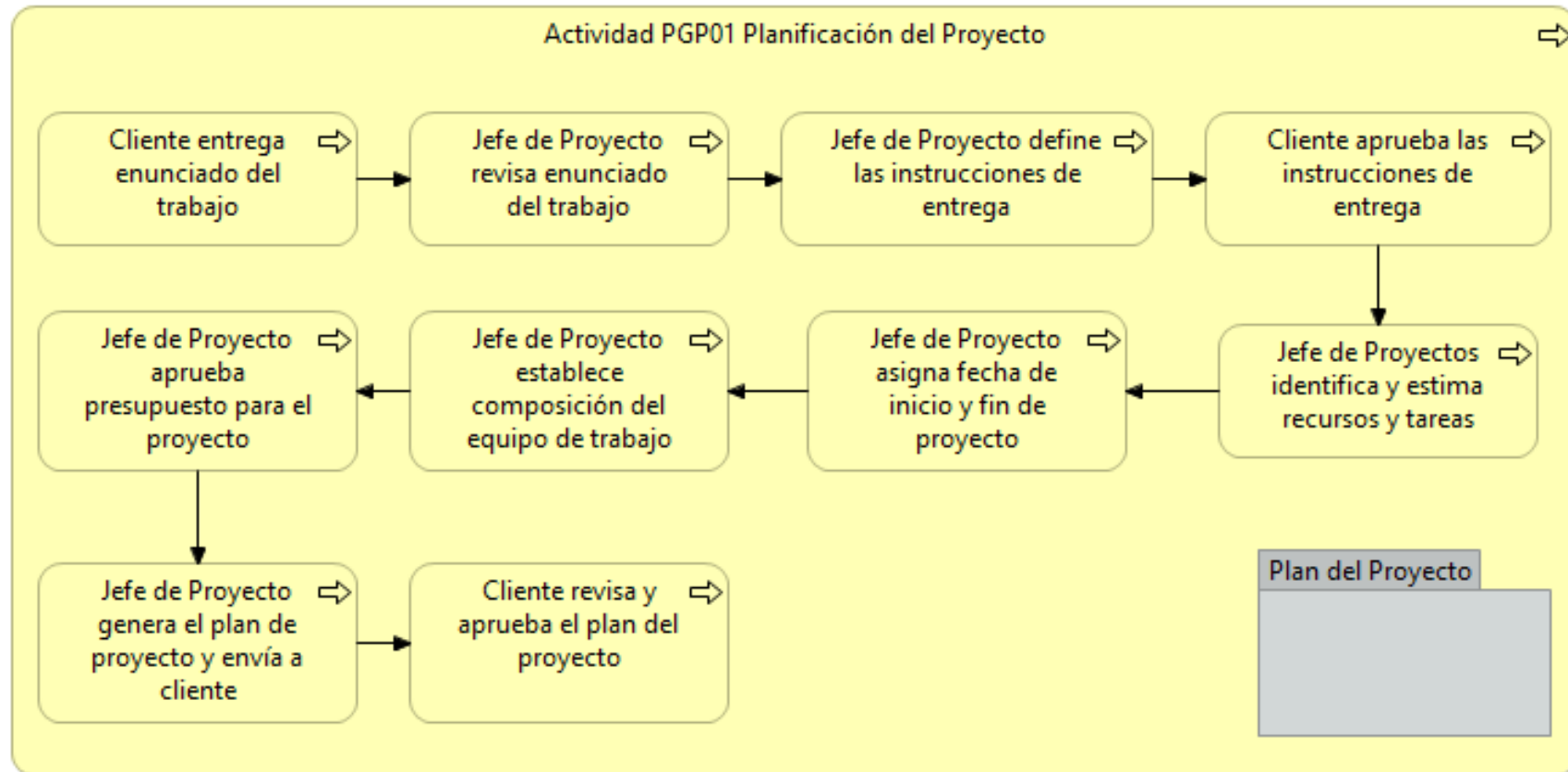


Figura 39, Actividad PGP01 Planificación del Proyecto

Fuente, elaboración propia.

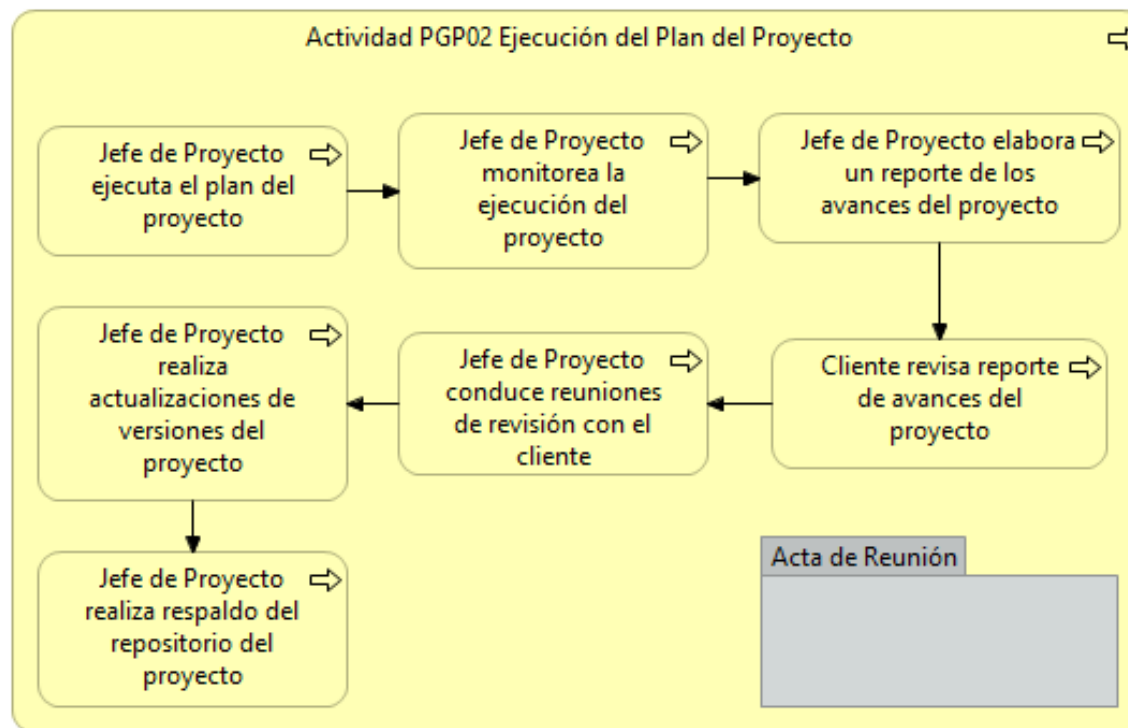


Figura 40, Actividad PGP02 Ejecución del Plan del Proyecto

Fuente, elaboración propia.

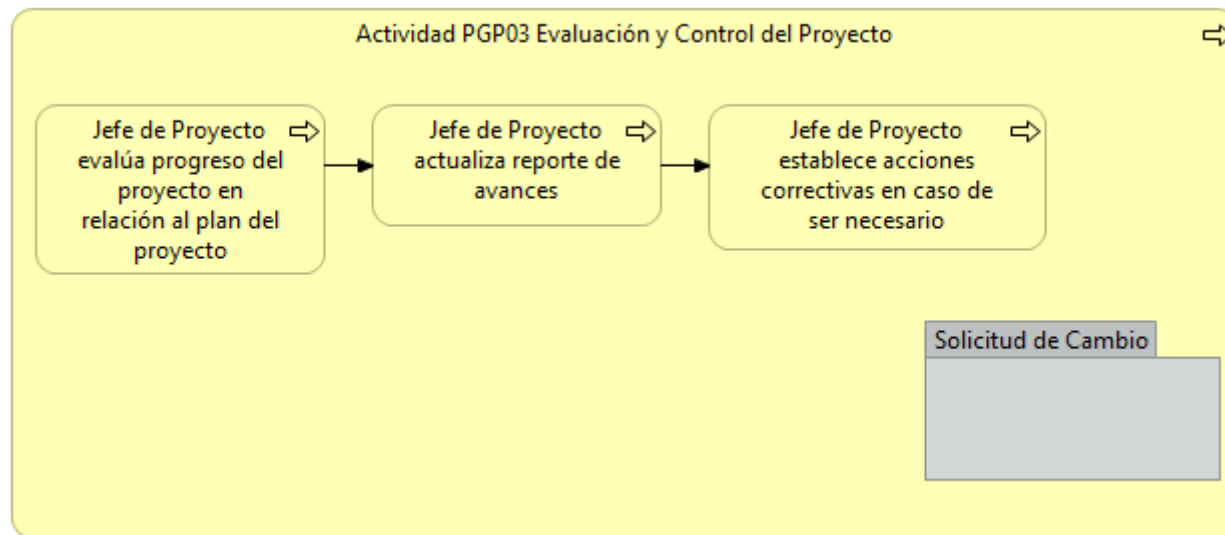


Figura 41, Actividad PGP03 Evaluación y Control del Proyecto

Fuente, elaboración propia.

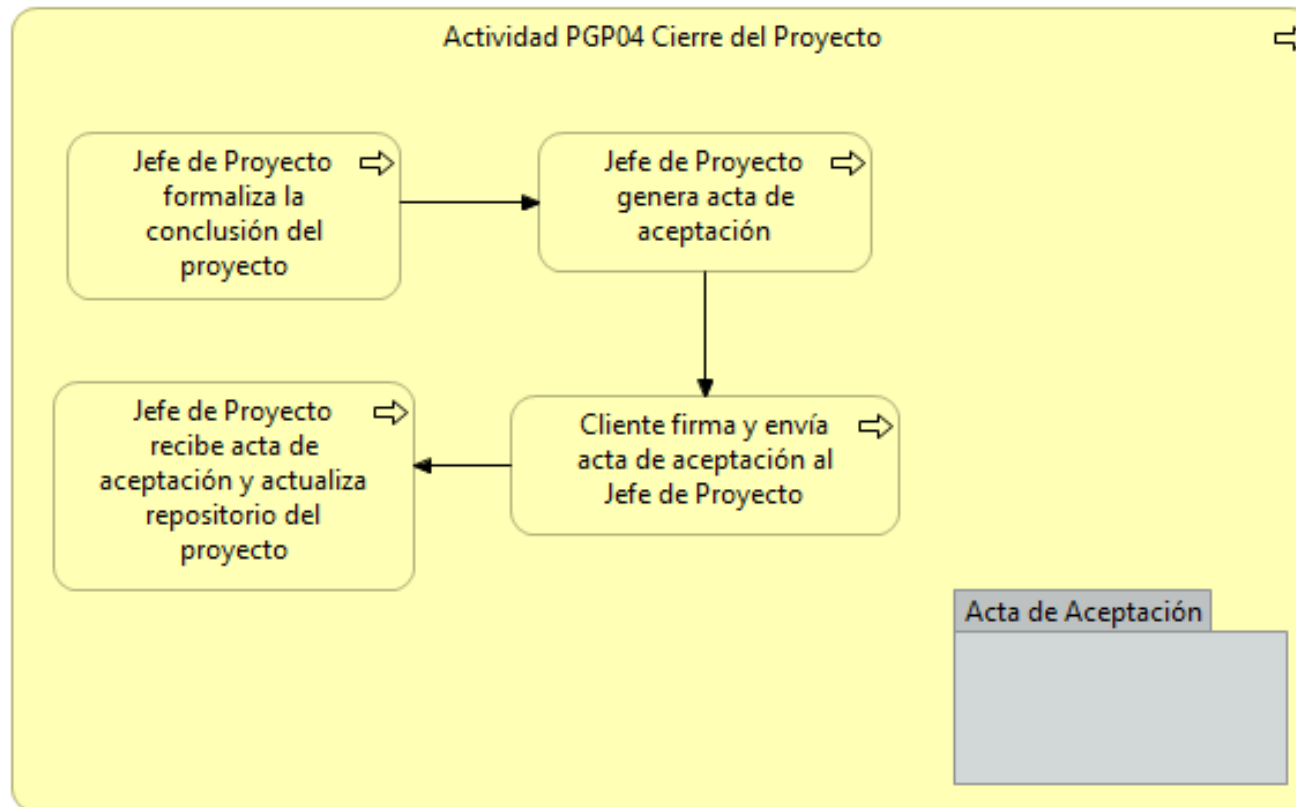


Figura 42, Actividad PGP04 Cierre del Proyecto

Fuente, elaboración propia.

El Proceso de Implementación de Software propuesto para este Modelo de construcción de software comprende cuatro (04) actividades a las que se ha denominado:

- a. PIS01 Iniciación y Análisis de Requisitos de Software
- b. PIS02 Arquitectura, Diseño y Construcción de Software
- c. PIS03 Integración y Pruebas de Software
- d. PIS04 Entrega del Producto

Todos los productos de trabajo utilizados por el proceso de implementación de software se agrupan en los grupos de productos de entrada, internos y de salida. Este Proceso de Implementación de Software para el Modelo de Construcción de Software propuesto se encuentra modelado en la Figura 43.

Asimismo, los Objetivos del Proceso de Implementación de Software propuesto para este Modelo de Construcción de Software tiene en su composición siete (07) objetivos a saber y que se encuentran modelados en la Figura 44.

Los siete objetivos del proceso SI se citan a continuación:

Tabla 18

Objetivos del proceso de Implementación de Software del Perfil Básico ISO/IEC 29110.

Objetivo	Descripción
PIS.O1	El cumplimiento del Plan de Proyecto vigente, se cumplirá con las tareas y actividades propuestas.
PIS.O2	Los requisitos de software se definen, analizan para verificar su exactitud y capacidad de prueba, son aprobados por el Cliente, se establecen como referencia y se comunican.

- PIS.O3 Se realiza un bosquejo de la arquitectura detallada del software y se establece como base. Se trazan los elementos de software internas y externas de los mismos.
- PIS.O4 Se producen los componentes de software definidos por el diseño. Las pruebas unitarias se definen y realizan para verificar la coherencia con los requisitos y el diseño.
- PIS.O5 Integramos el software que se ha fabricado con sus componentes y se constata con las pruebas y procedimientos de prueba. Se registran los resultados en un informe de pruebas.
- PIS.O6 Una Configuración de Software, que cumple con la Especificación de Requisitos acordada con el Cliente, que incluye documentación de usuario, operación y mantenimiento, se integra, se establece como base y se almacena en el Repositorio del Proyecto.
- PIS.O7 Cada tarea de verificación y validación de cada producto de los trabajos requeridos se realizan teniendo en cuenta los parámetros precisados para alcanzar una congruencia entre los productos de entrada y salida en cada actividad.

Fuente, elaboración propia

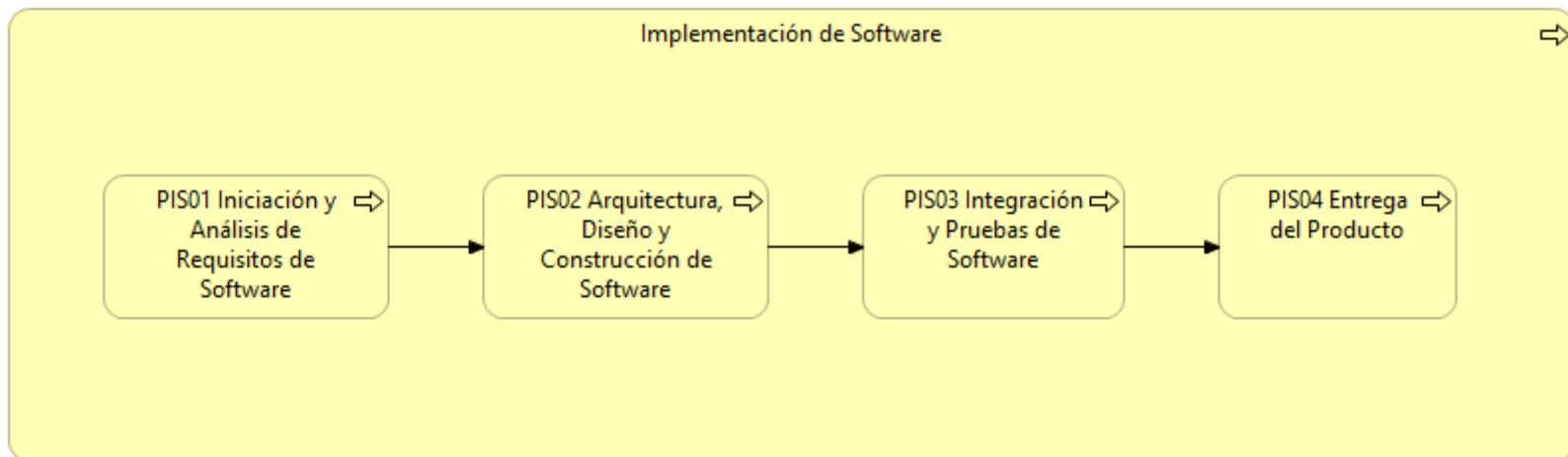


Figura 43, Vista del Proceso de Implementación de Software

Fuente, elaboración propia.

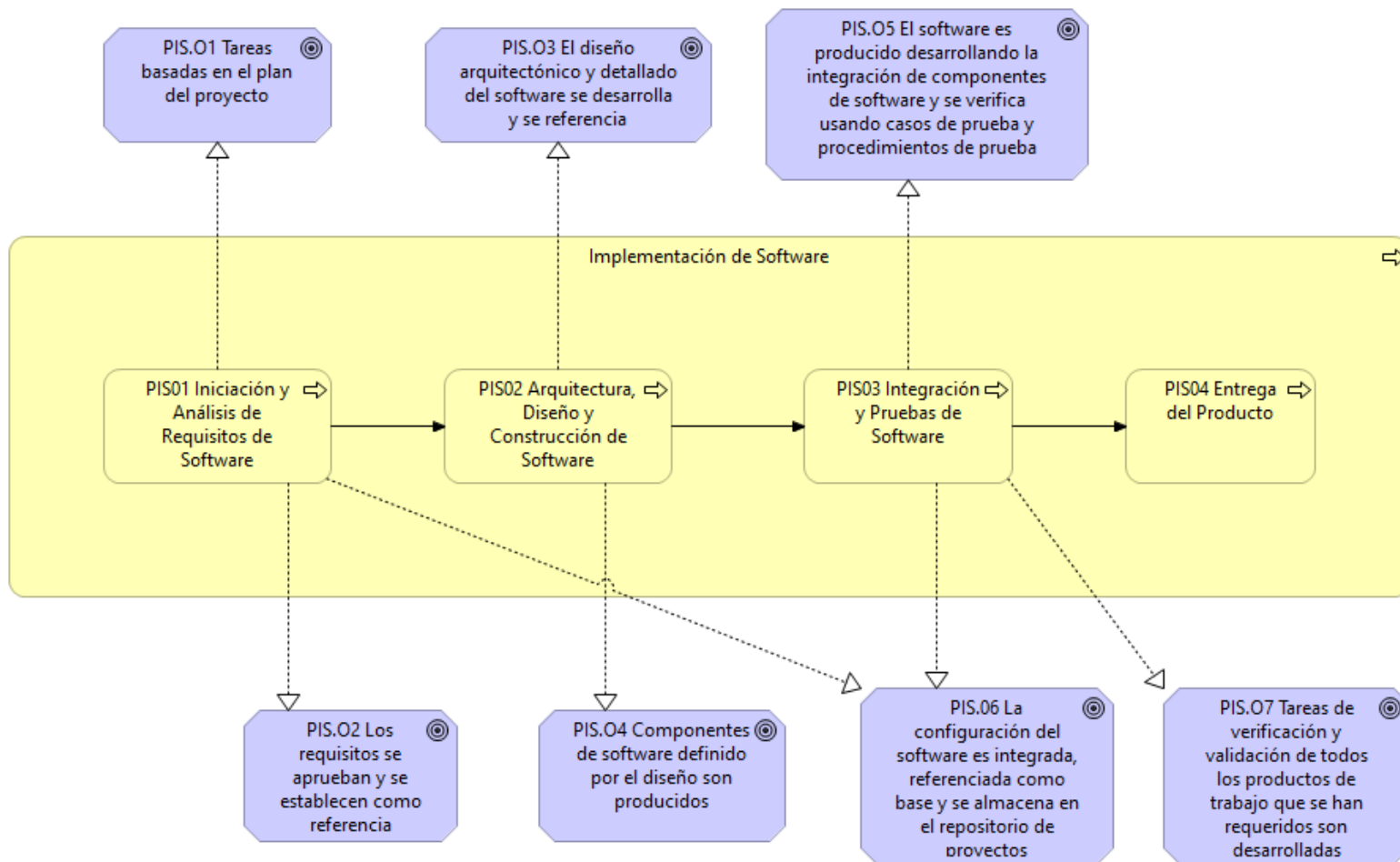


Figura 44, Objetivos de la vista del Proceso de Implementación del Software

Fuente, elaboración propia.

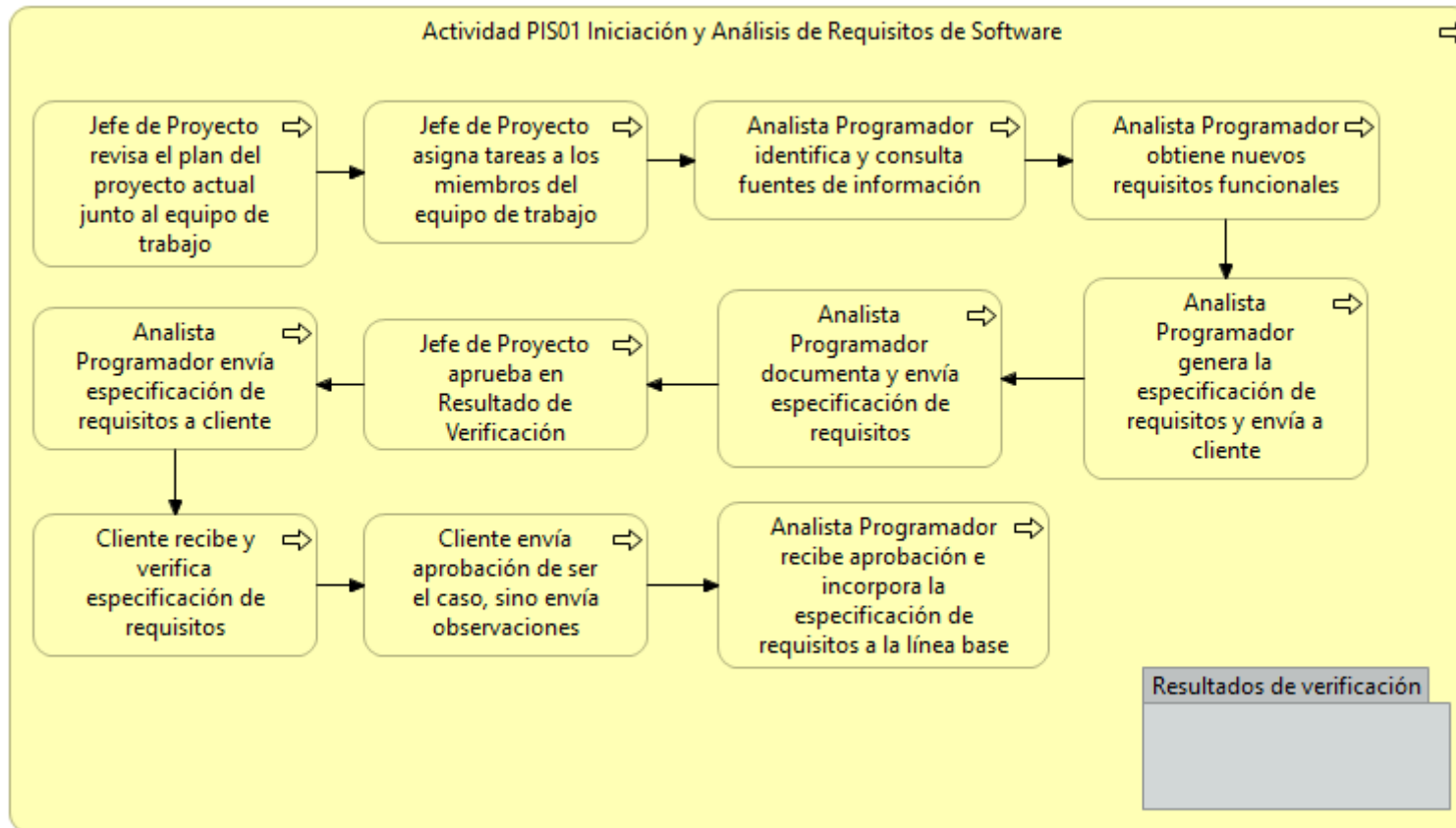


Figura 45, Actividad PIS01 Iniciación y Análisis de Requisitos de Software

Fuente, elaboración propia.

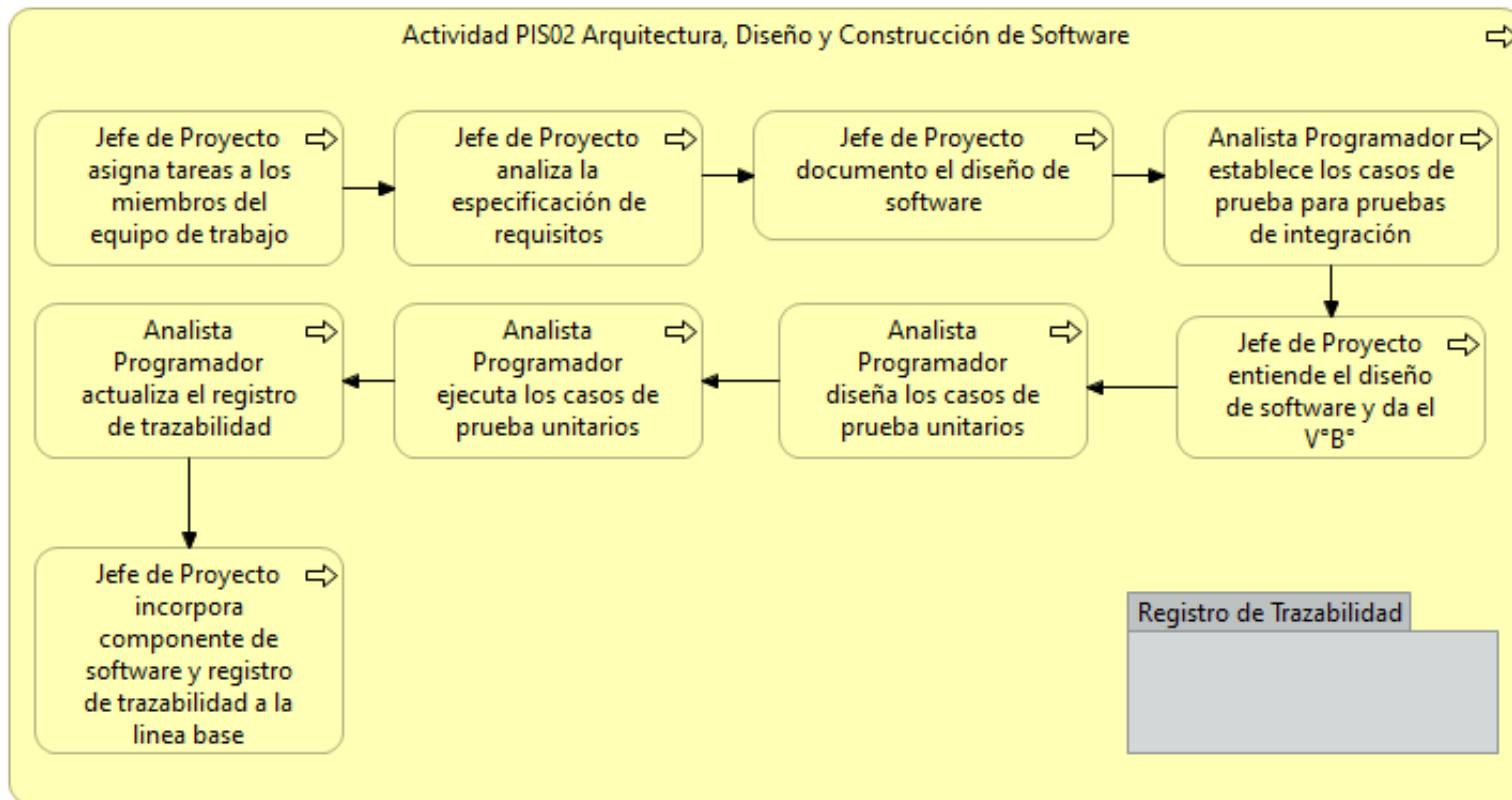


Figura 46, Actividad PIS02 Arquitectura, Diseño y Construcción de Software

Fuente, elaboración propia.

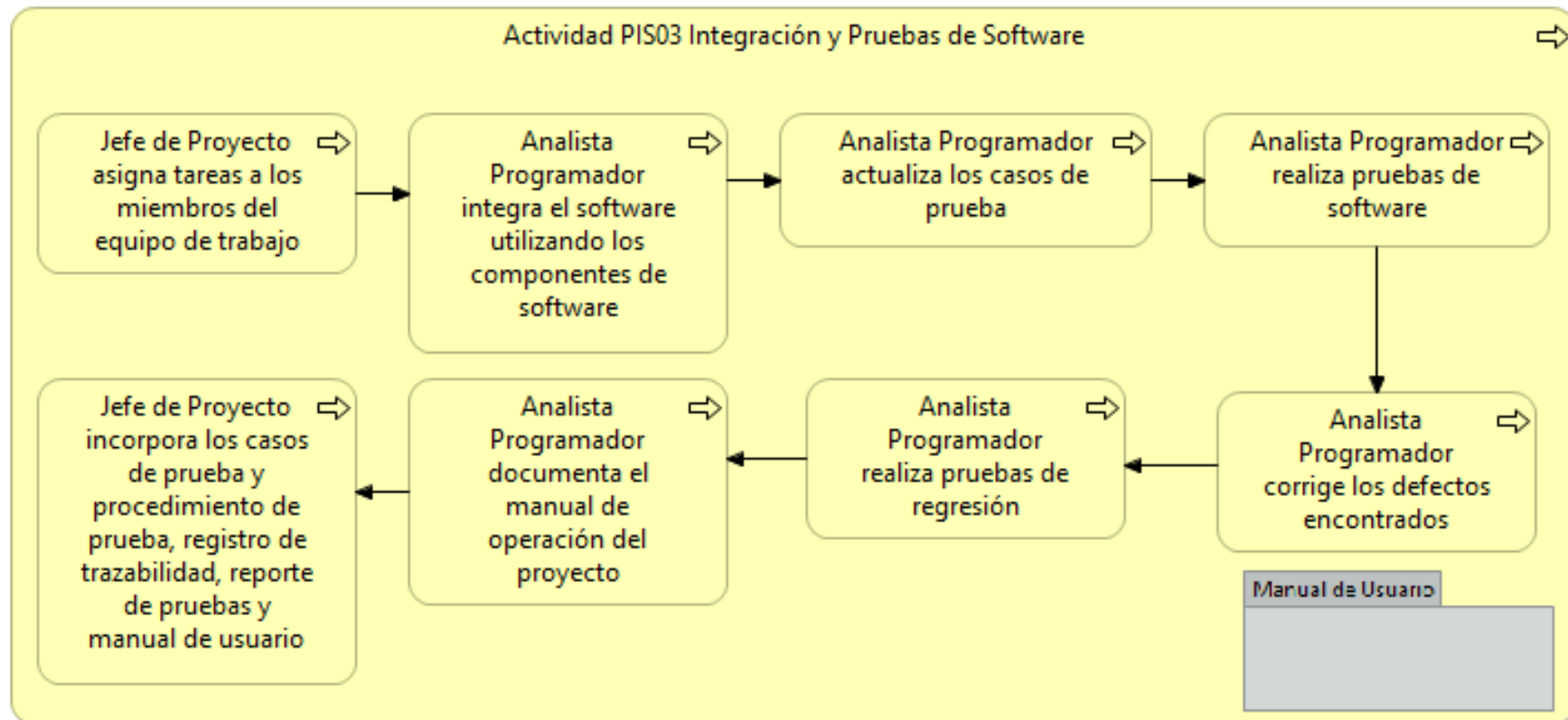


Figura 47, Actividad PIS03 Integración y Pruebas de Software

Fuente, elaboración propia.

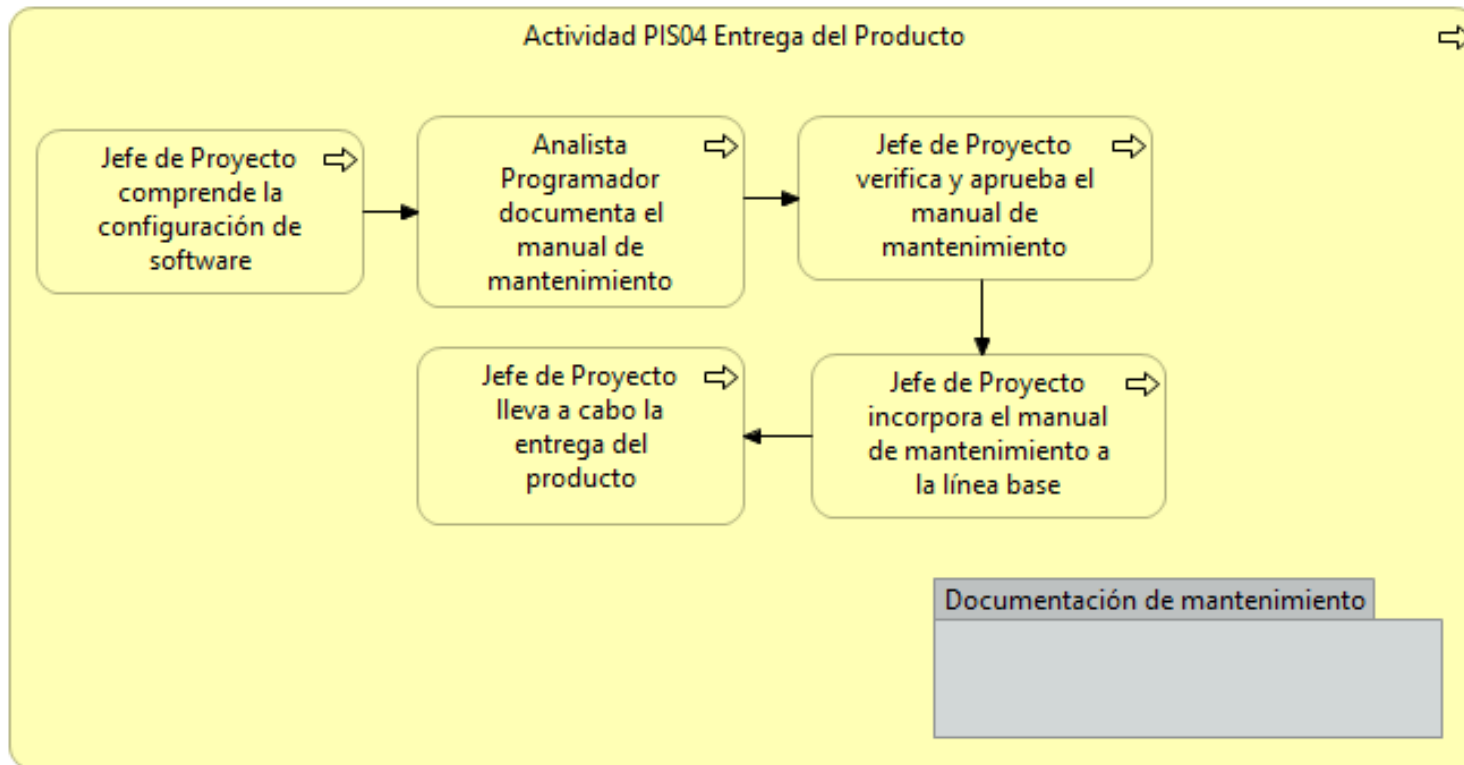


Figura 48, Actividad PIS04 Entrega del Producto

Fuente, elaboración propia.

Posteriormente, para lograr el presente objetivo de validar el presente modelo LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC se solicitó el juicio de 3 expertos con título de Ingeniero de Sistemas que, como característica primordial, cuenten con vasta experiencia en el desarrollo de software a medida. Dichos expertos se encuentran especificados en la tabla siguiente:

Tabla 19

Expertos para validación de instrumento.

N°	Apellidos y Nombres	Título	Grado Académico
1	Campos Perez Jahaira Zuleika	Ingeniera de Sistemas	Maestra en Gerencia en de Tecnología de Información y Comunicaciones
2	Salazar Carranza Jhonatan David	Ingeniero de Sistemas	Maestro (c) en Dirección de de Sistemas y Tecnologías de la Información
3	Echegaray Jorge Luis	Ingeniero de Sistemas Cómputo	de y Maestro (c) en Ingeniería de Software

Fuente, elaboración propia

A dichos expertos se les envió mediante email una copia de un formato de juicio de expertos que consistió en dar su visto bueno en cuanto a los siguientes indicadores y criterios para la evaluación del instrumento: claridad, objetividad, actualidad, organización, suficiencia, intencionalidad, consistencia, coherencia, metodología, pertinencia. Dicha validación, constó con la siguiente valoración: Valoración: Deficiente - [50-200], Baja - [250-400], Regular - [450-600], Buena: [650-800], Muy Buena - [850-1000]. Los aspectos de validación de dicho juicio de expertos se muestran a continuación:

Tabla 20

Aspectos de validación del juicio de expertos.

Tabla 21**Resultados validación de expertos**

Experto	CLARIDAD	OBJETIVIDAD	ACTUALIDAD	ORGANIZACIÓN	SUFICIENCIA	INTENCIONALIDAD	CONSISTENCIA	COHERENCIA	METODOLOGÍA	PERTINENCIA
Experto 01	95	95	95	95	90	90	90	95	95	95
Experto 02	95	95	95	95	95	95	95	95	95	95
Experto 03	95	90	95	90	95	90	95	95	95	95

Fuente, elaboración propia.

Tabla 22**Promedio de validación del modelo**

Experto	Promedio	Promedio Final
Experto 01	93,50	
Experto 02	95,00	94,00
Experto 03	93,50	

Fuente, elaboración propia.

Por tanto, el presente modelo de procesos para la construcción de software de calidad en microempresas peruanas desarrolladoras de software obtuvo una valoración promedio de 94,00 con lo que cuenta con el estatus de “MUY BUENA PROPUESTA” según el criterio de los expertos.

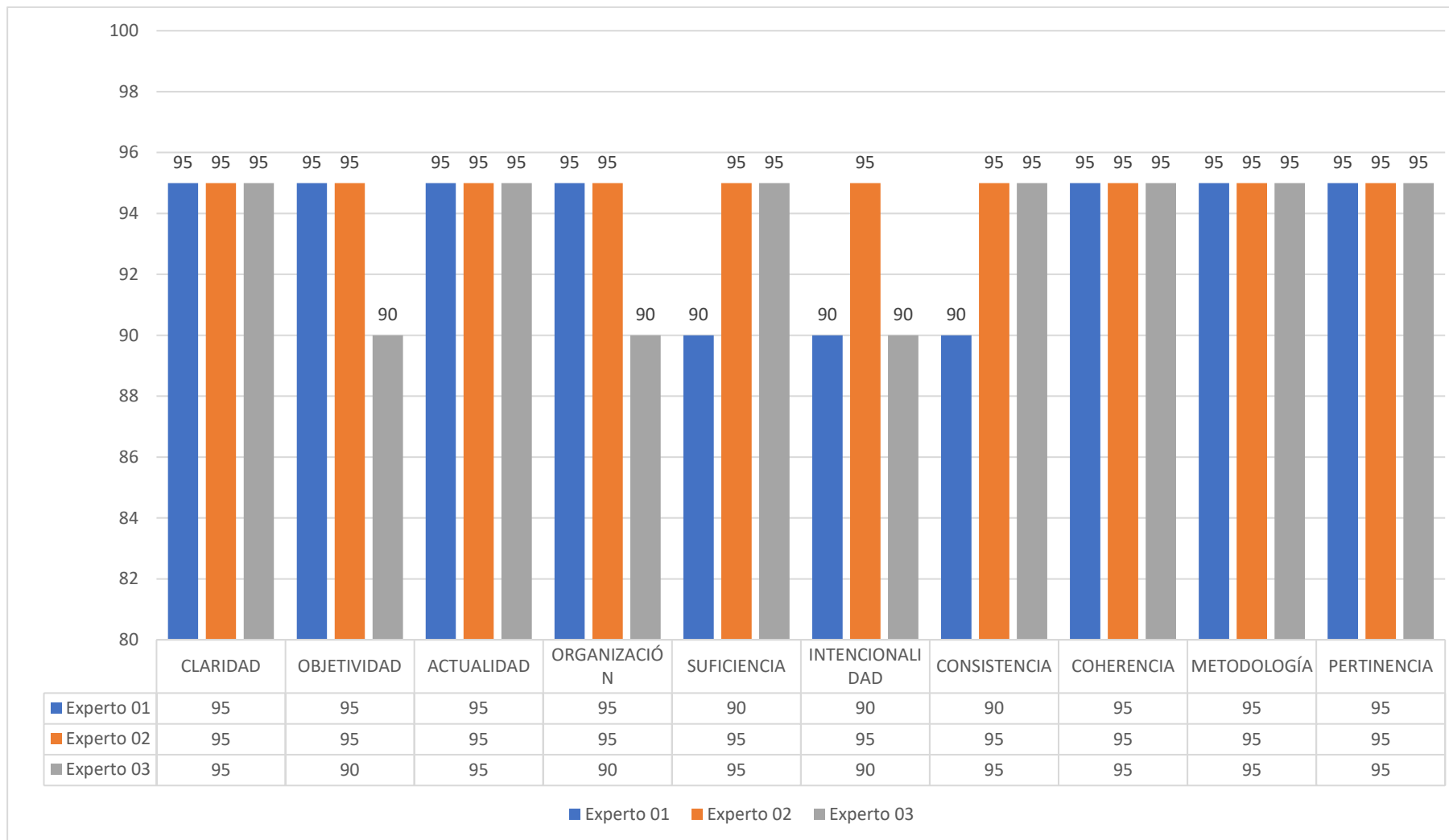


Figura 49, Evaluación general del modelo de procesos

Fuente, elaboración propia.

IV. CONCLUSIONES Y RECOMENDACIONES

3.4. Conclusiones.

Se caracterizaron los modelos de procesos para la construcción de software de calidad determinándose que, la Norma ISO/IEC 29110 y el Modelo MoProSoft fueron los más convenientes dado que, las estructuras de sus marcos metodológicos permiten la usanza por parte de pequeñas organizaciones constructoras de software.

Se analizaron la Norma ISO/IEC 29110 y el Modelo MoProSoft determinándose que, ambas metodologías poseían una hoja de ruta muy equivalente entre los procesos involucrados en la construcción de software, contemplando la Gestión de Procesos, la Gestión de Negocios y la Gestión de Proyectos.

Se diseñó un modelo de procesos para la construcción de software de calidad, el mismo que se basó en la Norma ISO/IEC 29110 y el Modelo MoProSoft, el cual se desplegó en dos procesos principales, la Gestión de Proyectos y la Implementación de Software, teniendo como soporte para el modelado a la herramienta de código abierto ArchiMate, con lo que se mejoró la calidad del software construido en los proyectos de la empresa Sistema Inteligente ERP SAC.

El modelo de construcción de software propuesto, fue validado mediante juicio de expertos quienes dieron su veredicto positivo dado que, se cumplía con los requerimientos claves que las microempresas constructoras de software demandan, determinándolo como “Muy buena propuesta”.

3.5. Recomendaciones.

Se recomienda a aquellas pequeñas organizaciones del rubro de la construcción de software, llevar a la praxis el presente modelo dado que, se encuentra enmarcado en la Norma ISO/IEC 29110 y el Modelo MoProSoft, lo que les permitirá garantizar la coherencia entre sus procesos constructivos y la calidad de los mismos.

Se recomienda a los gerentes y directivos de las pequeñas organizaciones del rubro de la construcción de software, mostrar el compromiso y liderazgo para la adaptabilidad e implementación de los modelos de calidad de procesos en sus procesos constructivos de software.

Se recomienda a los coordinadores de proyectos de construcción de software, el empleo del perfil básico de la ISO/IEC 29110 dado que, es el perfil más antiguo y el único con el que se pueden certificar las pequeñas organizaciones.

Se recomienda a futuros investigadores, emplear el lenguaje de modelado ArchiMate para modelar la metodología y/o procesos para la construcción de un sistema o producto de software dado que, cuenta con herramientas de notación y modelado que reducen significativamente la complejidad de la implementación y aumenta la comprensión de los modelos existentes.

REFERENCIAS.

- Abrahamsson, P., Oza, N., & Siponen, M. (2010). Agile software development methods: a comparative review. In Springer, Berlin, Heidelberg. *Agile software development*, 31-59. Obtenido de https://link.springer.com/chapter/10.1007/978-3-642-12575-1_3
- Al-Herwi, S. (2019). What are SMEs? *The International Journal of Management Education*. Obtenido de <https://www.researchgate.net/publication/332539278>
- Altwies, D., & Preston, J. (2017). *Achieve PMP Exam Success: A Concise Study Guide for the Busy Project Manager* (Sexta ed.). J. Ross Publishing.
- Amable, M. (2015). Propuesta de un proceso de investigación cuantitativa. Aplicación en la caracterización de las mypes productoras de software. *Revista Interfases*, 8, 71-92. Obtenido de <http://repositorio.ulima.edu.pe/handle/ulima/2535>
- Amable, M. (2017). Propuesta de Modelo de mejora para MYPEs productoras de software. *Revista Interfases*, 10, 57-73. Obtenido de <http://repositorio.ulima.edu.pe/handle/ulima/5405>
- Amable, M., & Millones, R. (2019). Uso de modelos de calidad en las mypes productoras de software de Lima. *Revista de Ingeniería Industrial*(37). Obtenido de <https://www.researchgate.net/publication/337451066>
- Amable, M., Millones, R., & Checa, R. (2015). Calidad de software en las mypes productoras de software en Lima. *Memoria COMTEL 2015 - VII Congreso Internacional de Computación y Telecomunicaciones* (págs. 133-139). Lima, Perú: Fondo Editorial - Universidad Inca Garcilaso de la Vega. Obtenido de <http://www.comtel.pe/memoriacomtel/COMTEL2015.pdf>
- Arboleda, H., Paz, A., & Casallas, R. (2013). Metodología para implantar el Modelo Integrado de Capacidad de Madurez en grupos pequeños y emergentes. *Estudios Gerenciales*, 29(127), 177-188. Obtenido de <https://www.sciencedirect.com/science/article/pii/S0123592313000077>
- Bayona, S., Chamilco, J., & Pérez, D. (2019). Software Process Improvement: Requirements Management, Verification and Validation. *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)* (págs. 1-5). Coimbra: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/8760896>

- Bernal, C. (2010). *Metodología de la investigación*. Bogotá, Colombia: Pearson Education SA.
- Boehm, B., & Turner, R. (2003). *Balancing agility and discipline: A guide for the perplexed*. Boston: Addison-Wesley Professional.
- Burton, J. (23 de Agosto de 2017). *At least 800,000 struggling families face crisis after sub-prime loan firm Provident Financial goes into meltdown*. Obtenido de Daily Mail: <https://www.dailymail.co.uk/news/article-4814606/800-000-families-face-crisis-sub-prime-loan-firm-tanks>
- Callejas, M., Alarcón, A., & Álvarez, A. (2017). Modelos de calidad del software, un estado del arte. *Entramado*, 13(1), 236-250. Obtenido de <https://revistas.unilibre.edu.co/index.php/entramado/article/view/428>
- Castro, F., Kellison, J., Boyd, S., & Kopak, A. (2010). A methodology for conducting integrative mixed methods research and data analyses. *Journal of mixed methods research*, 4(4), 342-360. Obtenido de <https://doi.org/10.1177/1558689810382916>
- Chevers, D. (2017). Software process improvement: awareness, use, and benefits in Canadian software development firms. *Revista de Administração de Empresas*, 57(2), 170-177. Obtenido de <https://doi.org/10.1590/s0034-759020170206>
- Chevers, D., Mills, A., Duggan, E., & Moore, S. (2016). An evaluation of software development practices among small firms in developing countries: A test of a simplified software process improvement model. *Journal of Global Information Management*, 24(3), 45-70. Obtenido de <https://www.igi-global.com/gateway/article/163453>
- Chevers, D., Mills, A., Duggan, E., & Moore, S. (2017). Toward a Simplified Software Process Improvement Framework for Small Software Development Organizations. *Journal of Global Information Technology Management*, 20(2), 110-130. Obtenido de <https://doi.org/10.1080/1097198X.2017.1321356>
- Clarke, P., & O'Connor, R. (2012). The situational factors that affect the software development process: Towards a comprehensive reference framework. *Information and Software Technology*, 54(5), 433-447. Obtenido de <https://doi.org/10.1016/j.infsof.2011.12.003>

- Clarke, P., & O'Connor, R. (2013). An empirical examination of the extent of software process improvement in software SMEs. *Journal of Software: Evolution and Process*, 25(9), 981-998. Obtenido de <https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.1580>
- CMMI. (2010). *CMMI for Development, Version 1.3. CMU/SEI-2010-TR-033*. Software Engineering Institute. Pensilvania, EEUU: Carnegie Mellon University. Obtenido de https://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15287.pdf
- Coleman, G., & O'Connor, R. (2008). Investigating software process in practice: A grounded theory perspective. *Journal of Systems and Software*, 81(5), 772-784. Obtenido de <https://www.sciencedirect.com/science/article/pii/S0164121207001811>
- Commission, E. (28 de Marzo de 2014). *What is an SME?* Obtenido de Business-friendly environment: https://ec.europa.eu/growth/smes/business-friendly-environment/sme-definition_en
- Conrad, E., Misener, S., & Feldman, J. (2016). *Eleventh Hour CISSP®: Study Guide*. Syngress. (Tercera ed.). Syngress.
- Coque, S., Jurado, V., & Avendaño, A. &. (2018). Análisis de experiencias de mejora de procesos de desarrollo de software en PYMEs. *Revista Ciencia UNEMI*, 10(25), 13-24. Obtenido de <https://www.researchgate.net/publication/326027868>
- Díaz, A., De Jesús, C., Melendez, K., & Dávila, A. (2016). The ISO/IEC 29110 Implementation on two Very Small Software Development Companies in Lima. Lessons Learned. *IEEE Latin America Transactions*, 14(5), 2504-2510. Obtenido de <https://ieeexplore.ieee.org/document/7530452>
- Espejo, A., Bayona, S., & Pastor, C. (2016). Quality assurance in the software development process using CMMI, TSP and PSP. *Revista Ibérica de Sistemas e Tecnologias de Informação*(20), 62-77. Obtenido de <http://dx.doi.org/10.17013/risti.20.62-77>
- Frischknecht, C. (31 de Octubre de 2017). *Software Fail Watch, Q3 2017: Software Fails Are Bigger Than Ever* . Obtenido de Tricentis: <https://www.tricentis.com/blog/software-fail-watch-q3-2017/>

- Gagnon, Y. (2010). *The case study as research method: A practical handbook* (Primera ed.). Québec: Presse de l'Université du Québec.
- Hallowell, M., & Gambatese, J. (2010). Qualitative research: Application of the Delphi method to CEM research. *Journal of construction engineering and management*, 136(1), 99-107. Obtenido de [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0000137](https://doi.org/10.1061/(ASCE)CO.1943-7862.0000137)
- INEI. (2019). *Perú: Estructura Empresarial, 2018*. Lima: Instituto Nacional de Estadística e Informática. Obtenido de https://www.inei.gob.pe/media/MenuRecursivo/publicaciones_digitales/Est/Lib1703/libro.pdf
- ISO. (2017). *ISO/IEC/IEEE 12207:2017. Systems and software engineering - Software life cycle processes*. Obtenido de <https://www.iso.org/standard/63712.html>
- Kabaale, E., Wen, L., Wang, Z., & Rout, T. (2017). An Axiom Based Metamodel for Software Process Formalisation: An Ontology Approach. *International Conference on Software Process Improvement and Capability Determination* (págs. 226-240). Cham: Springer. Obtenido de <https://www.semanticscholar.org/paper/An-Axiom-Based-Metamodel-for-Software-Process-An-Kabaale-Wen/7c0bb08e919b0a8be3e3b0354f1cdaa3756aca5d>
- Khan, A., Keung, J., Fazal-e-Amin, I., & Abdullah-Al-Wadud, M. (2017). SPIIMM: Toward a Model for Software Process Improvement Implementation and Management in Global Software Development. *IEEE Access*, 5(1), 13720-13741. Obtenido de <https://ieeexplore.ieee.org/abstract/document/7983363>
- King, R., Hooper, B., & Wood, W. (2011). Using bibliographic software to appraise and code data in educational systematic review research. *Journal Medical Teacher*, 33(9), 719-723.
- Krasner, H. (2018). *The Cost of Poor Quality Software in the US: A 2018 Report*. Milford: Consortium for IT Software Quality. Obtenido de <https://www.it-cisq.org/the-cost-of-poor-quality-software-in-the-us-a-2018-report/>
- Laporte, C., Alexandre, S., & O'Connor, R. (2008). A software engineering lifecycle standard for very small enterprises. *European Conference on Software*

- Process Improvement*, 16, 129-141. Obtenido de https://link.springer.com/chapter/10.1007/978-3-540-85936-9_12
- Laporte, C., Muñoz, M., & Gerançon, B. (2017). The education of students about ISO/IEC 29110 software engineering standards and their implementations in very small entities. *IEEE Canada International Humanitarian Technology Conference (IHTC)* (págs. 94-98). Toronto, Ontario, Canada: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/8058208>
- Larrucea, X., O'Connor, R., Colomo, R., & Laporte, C. (2016). Software Process Improvement in Very Small Organizations. *IEEE Software*, 33(2), 85-89. Obtenido de <https://ieeexplore.ieee.org/document/7420471>
- López, W., Bernuy, W., & Pando, B. (2019). Bringing a Micro Enterprise towards the ISO 29110. *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)* (págs. 1-6). Coimbra: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/8760865>
- Mahmoud, Z., Solyman, A., & Elhag, A. (2019). Harmonized Software Quality Improvement Models for Sudanese SME Based on CMMI. *2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)* (págs. 1-6). Jartum: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/9071148>
- Main, E. (15 de Diciembre de 2016). *Home Office under fire after student visa fraud case*. Obtenido de BBC: <https://www.bbc.com/news/uk-38236852>
- Maki, S., & Sage, A. (19 de Marzo de 2018). *Self-driving Uber car kills Arizona woman crossing street*. Obtenido de Reuters: <https://www.reuters.com/article/us-autos-selfdriving-uber-idUSKBN1GV296>
- Mall, R. (2014). *Fundamentals of software engineering* (Cuarta ed.). Kharagpur: PHI Learning Private Limited.
- McFall, D., Wilkie, F., McCaffery, F., Lester, N., & Sterrit, R. (2003). Software processes and process improvement in Northern Ireland. *16th International Conference on Software and Systems Engineering and their Applications*, (págs. 1-10). Paris. Obtenido de <https://eprints.dkit.ie/147/>
- Mishra, D., & Mishra, A. (2009). Software process improvement in SMEs: A comparative view. *Computer Science and Information Systems*, 6(1), 111-140. Obtenido de <https://doi.org/10.2298/CSIS0901111M>

- Moll, R. (2013). *Being prepared—A bird's eye view of SMEs and risk management. ISO Focus+*. Ginebra, Suiza: International Organization for Standardization.
- Moreno, E., Sánchez, M., Colomo, R., & de Amescua, A. (2014). Towards measuring the impact of the ISO/IEC 29110 standard: a systematic review. *European Conference on Software Process Improvement*. 425, págs. 1-12. Heidelberg: IEEE. Obtenido de https://link.springer.com/chapter/10.1007/978-3-662-43896-1_1
- Muñoz, M., Mejía, J., & Laporte, C. (2018). Implementación del Estándar ISO/IEC 29110 en Centros de Desarrollo de Software de Universidades Mexicanas: Experiencia del Estado de Zacatecas - Medellín, Columbia. *Conference: XI Seminario Internacional de Ciencias de la Computación*. 29, págs. 43-54. Medellín, Colombia: RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação. Obtenido de <https://www.researchgate.net/publication/328198963>
- Muñoz, M., Peña, A., Mejía, J., Gasca, G., Gómez, M., & Laporte, C. (2019). A Comparative Analysis of the Implementation of the Software Basic Profile of ISO/IEC 29110 in Thirteen Teams That Used Predictive Versus Adaptive Life Cycles. *26th European Conference on Systems, Software and Services Process Improvement, EuroSPI 2019*. 1060. Edinburgh, Reino Unido: Springer Verlag. Retrieved from <https://www.researchgate.net/publication/335700159>
- Noreña, A., Alcaraz, N., Rojas, J., & Rebolledo, D. (2012). Aplicabilidad de los criterios de rigor y éticos en la investigación cualitativa. *Revista Científica Aquichan*, 263-274.
- Ñaupas, H., Mejía, E., Novoa, E., & Villagómez, A. (2014). *Metodología de la investigación. Cuantitativa - Cualitativa y Redacción de la Tesis*. (Cuarta ed.). Bogotá, Colombia: Ediciones de la U.
- OECD. (2011). Regions and Innovation Policy, Organization for Economic Cooperation and Development. *Reviews of Regional Innovation*. Obtenido de <http://www.oecd.org/innovation/oecdreviewsofregionalinnovationregionsandinnovationpolicy.htm>

- Oktaba, H. (2005). *Modelo de Procesos para la Industria de Software: MoProSoft*. Ciudad de México: Asociación Mexicana para la Calidad en Ingeniería de Software. Obtenido de <https://www.researchgate.net/publication/267028000>
- Pino, F., García, F., Piattini, M., & Oktaba, H. (2016). A research framework for building SPI proposals in small organizations: the COMPETISOFT experience. *Software Quality Journal*, 24(3), 489-518. Obtenido de <https://link.springer.com/article/10.1007/s11219-015-9278-2>
- Pressman, R. (2010). *Software engineering: a practitioner's approach* (Séptima ed.). Nueva York: The McGraw-Hill.
- PromPerú. (2011). *Perú Portafolio de Software 2011*. Comisión de Promoción del Perú para la Exportación y el Turismo - PromPerú, Lima, Perú.
- Quinn, T. (3 de Agosto de 2017). *Major HSE computer glitch means 25,000 Irish patients may need new scans*. Obtenido de Irish Daily Mirror: <https://www.irishmirror.ie/news/irish-news/health-news/major-hse-computer-glitch-means-10925730>
- Richardson, I., & Von Wangenheim, C. (2007). Guest editors' introduction: Why are small software organizations different? *IEEE Software*, 24(1), 18-22. Obtenido de <https://ieeexplore.ieee.org/document/4052546>
- Sánchez, M., Colomo, R., de Amescua, A., & O'Connor, R. (2017). The Route to Software Process Improvement in Small and Medium-sized Enterprises. En M. Kuhrmann, J. Münch, I. Richardson, A. Rausch, & H. Zhang, *Managing Software Process Evolution: Traditional, Agile and Beyond - How to Handle Process Change* (págs. 109-136). Suiza: Springer International Publishing. Obtenido de http://dx.doi.org/10.1007/978-3-319-31545-4_7
- Sánchez, S., & Viera, D. (2019). Mapping between CMMI-DEV v1.3 and ISO/IEC 90003:2014. *2019 International Conference on Information Systems and Software Technologies (ICI2ST)* (págs. 134-140). Quito: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/8940387>
- Singh, R. (1996). International standard ISO/IEC 12207 software life cycle processes. *Software Process: Improvement and Practice 2.1*, 2(1), 35-50. Obtenido de [https://onlinelibrary.wiley.com/doi/abs/10.1002/\(SICI\)1099-1670\(199603\)2:1%3C35::AID-SPIP29%3E3.0.CO;2-3](https://onlinelibrary.wiley.com/doi/abs/10.1002/(SICI)1099-1670(199603)2:1%3C35::AID-SPIP29%3E3.0.CO;2-3)

- Sommerville, I. (2011). *Software engineering* (Novena ed.). Massachusetts: Pearson Education.
- Suárez, D., & León, G. (2019). Las PyME de desarrollo de software. Modelos de mejora de sus procesos en Latinoamérica. *Revista Espacios*, 40(28), 9. Obtenido de <https://www.revistaespacios.com/a19v40n28/19402809.html>
- SUNAT. (19 de Febrero de 2019). *Iniciando mi negocio*. Obtenido de Emprender SUNAT: <http://emprender.sunat.gob.pe/que-beneficios-tengo>
- Suteeca, K. (2020). A Software Process Gap Analysis Methodology for Very Small Entity. *2020 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)* (págs. 190-193). Pattaya: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/9090745>
- Suteeca, K., & Ramingwong, S. (2016). A framework to apply ISO/IEC29110 on SCRUM. *2016 International Computer Science and Engineering Conference (ICSEC)* (págs. 1-5). Chiang Mai: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/7859884>
- Terrell, S. (2012). Mixed-Methods Research Methodologies. *Qualitative report*, 17(1), 254-280. Obtenido de <https://eric.ed.gov/?id=EJ973044>
- Valtierra, C., Muñoz, M., & Mejía, J. (2013). Characterization of software processes improvement needs in SMEs. *2013 International Conference on Mechatronics, Electronics and Automotive Engineering* (págs. 223-228). Morelos, México: IEEE. Obtenido de <https://ieeexplore.ieee.org/document/6713982>
- Vijayarathy, L., & Butler, C. (2016). Choice of software development methodologies: Do organizational, project, and team characteristics matter? *IEEE software*, 33(5), 86-94. Obtenido de <https://ieeexplore.ieee.org/abstract/document/7006383>
- Villarroel, R. (2016). *Aplicación de un Ciclo de Mejora para el Mejoramiento de los Procesos de Software en una Pequeña Empresa*. Curicó: Universidad Católica del Maule. Obtenido de <http://www.ubiobio.cl/miweb/webfile/media/182/presentacion/presentacionC hill%C3%A1n%202008.pdf>

Wildemuth, B. (2016). *Applications of social research methods to questions in information and library science*. (Segunda ed.). California, EEUU: Libraries Unlimited.

ANEXOS

Anexo 1. Resolución de aprobación del proyecto de investigación.

FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO
RESOLUCIÓN N°1315-2020/FIAU-USS

Pimentel, 17 de julio de 2020

VISTO:

El Acta de reunión N°1606-2020, de fecha 16 de junio de 2020 del Comité de investigación de la Escuela profesional de INGENIERIA DE SISTEMAS, para la ejecución : "DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC", presentado por el(los) tesista(s) HUANCAS MONTENEGRO JEINER y VARGAS MORENO, JORGE LUIS, del Programa de estudios INGENIERIA DE SISTEMAS, y;

CONSIDERANDO:

Que, de conformidad con la Ley Universitaria N° 30220 en su artículo 48º que a letra dice: "La investigación constituye una función esencial y obligatoria de la universidad, que la fomenta y realiza, respondiendo a través de la producción de conocimiento y desarrollo de tecnologías a las necesidades de la sociedad, con especial énfasis en la realidad nacional. Los docentes, estudiantes y graduados participan en la actividad investigadora en su propia institución o en redes de investigación nacional o internacional, creadas por las instituciones universitarias públicas o privadas.";

Que, de conformidad con el Reglamento de grados y títulos en su artículo 21° señala: "Los temas de trabajo de investigación, trabajo académico y tesis son aprobados por el Comité de Investigación y derivados a la facultad o Escuela de Posgrado, según corresponda, para la emisión de la resolución respectiva. El periodo de vigencia de los mismos será de dos años, a partir de su aprobación. En caso un tema perdiera vigencia, el Comité de Investigación evaluará la ampliación de la misma.

Que, de conformidad con el Reglamento de grados y títulos en su artículo 24º señala: La tesis es un estudio que debe denotar rigurosidad metodológica, originalidad, relevancia social, utilidad teórica y/o práctica en el ámbito de la escuela profesional. Para el grado de doctor se requiere una tesis de máxima rigurosidad académica y de carácter original. Es individual para la obtención de un grado; es individual o en pares para obtener un título profesional. Asimismo, en su artículo 25° señala: "El tema debe responder a alguna de las líneas de investigación institucionales de la USS S.A.C."

Que, en el Acta de reunión N°1606-2020 de fecha 16 de junio de 2020, del Comité de investigación de la Escuela profesional de INGENIERIA DE SISTEMAS, se indica entre los acuerdos la aprobación del Proyecto de tesis denominado "DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC" de la línea de investigación de INFRAESTRUCTURA, TECNOLOGÍA Y MEDIO AMBIENTE, a cargo de HUANCAS MONTENEGRO JEINER y VARGAS MORENO, JORGE LUIS en condición de estudiante, del Programa de estudios INGENIERIA DE SISTEMAS.

Estando a lo expuesto, y en uso de las atribuciones conferidas y de conformidad con las normas y reglamentos vigentes;

SE RESUELVE:

ARTÍCULO 1°: APROBAR, el Proyecto de denominado "DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC", perteneciente a la línea de investigación de INFRAESTRUCTURA, TECNOLOGÍA Y MEDIO AMBIENTE, a cargo de HUANCAS MONTENEGRO JEINER y VARGAS MORENO, JORGE LUIS, del Programa de estudios INGENIERIA DE SISTEMAS.

ARTÍCULO 2°: ESTABLECER, que la inscripción del Título de Proyecto de tesis se realice a partir de emitida la presente resolución y tendrá una vigencia de dos (02) años.

ARTÍCULO 3°: DEJAR SIN EFECTO, toda Resolución emitida por la Facultad que se oponga a la presente Resolución.

REGÍSTRESE, COMUNÍQUESE Y ARCHÍVESE



Dr. Mario Fernando Ramos Moscol
Decano - Facultad de Ingeniería,
Arquitectura y Urbanismo
UNIVERSIDAD SEÑOR DE SIPÁN SAC.



MBA. María Rosalía Salas Rivera
Secretaría Académica / Facultad de Ingeniería,
Arquitectura y Urbanismo
UNIVERSIDAD SEÑOR DE SIPÁN SAC.

Cc: Interesado, Archivo

Anexo 2. Carta de presentación del estudiante para realizar caso de estudio.



"Año de la Universalización de la Salud"

Pimentel, miércoles 07 de diciembre de 2020

Señor(a):
LOURDES CAROL ARIAS RAMOS
SUB GERENTE GENERAL
SISTEMA INTELIGENTE ERP SAC
Ciudad.-

ASUNTO:
Presentación de estudiante para realizar caso de estudio.

Es grato dirigirme a usted para expresarle el saludo institucional a nombre de la Escuela Profesional de Ingeniería de Sistemas, perteneciente a la Facultad de Ingeniería, Arquitectura y Urbanismo, de la Universidad Señor de Sipán, a la vez presentar a los estudiantes del X ciclo, VARGAS MORENO JORGE LUIS con código universitario 2150814473, e identificado con DNI 43432624, y HUANCAS MONTENEGRO JEINER, con código universitario 2161802403, e identificado con DNI 47059330, quienes recogerán información relevante en la institución que usted representa, como parte de su trabajo de INVESTIGACIÓN, aprobado con resolución N°1315-2020/FIAU-USS, del proyecto titulado "DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC".

Para ello, solicitamos su autorización, esperando que el estudiante cumpla con todos los requerimientos necesarios.

En espera de su atención a la presente, aprovecho la oportunidad para expresarle mi consideración y estima personal.

Cordialmente,

RÉCIBIDO 13 DIC 2020



Heber Ivan Mejia Cabrera
Mag. Ing. Heber Ivan Mejia Cabrera
Director (e) de la Escuela Profesional
de Ingeniería de Sistemas
UNIVERSIDAD SEÑOR DE SIPÁN S.A.C.

Lourdes Carol Arias Ramos
SISTEMA INTELIGENTE ERP S.A.C.
Lourdes Carol Arias Ramos
Sub-Gerente General

Anexo 3. Carta de aceptación de la institución para la recolección de datos.



"Año de la universalización de la salud"

Lima, 18 de Diciembre del 2020

Señor:

Mg. Ing. Heber Iván Mejía Cabrera

Director (e) de la Escuela Profesional de Ingeniería de Sistemas

Universidad Señor de Sipán

Asunto: Autorización para el recojo de información
cómo caso de estudio - INVESTIGACIÓN II

De mi especial consideración:

Es grato dirigirme a Usted, para expresarle mi cordial saludo en nombre de la empresa Sistema Inteligente ERP SAC, deseándole los mayores éxitos y a la vez dar respuesta al documento de fecha 07 de diciembre del 2020.

En este sentido **AUTORIZO** que, los estudiantes **HUANCAS MONTENEGRO JEINER** identificado con DNI N° 47059330 y **VARGAS MORENO JORGE LUIS** identificado con DNI N° 43432624, recojan información relevante para su trabajo de INVESTIGACIÓN, aprobado con Resolución N° 1315-2020/FIAU-USS, del proyecto titulado "DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC".

Seguro de contar con su interés, aprovecho la oportunidad para expresarle las muestras de mi especial consideración y estima.

Sin otro en particular, me despido con la mayor cordialidad.

Atentamente,

SISTEMA INTELIGENTE ERP S.A.C.


Lourdes Carol Arias Ramos
Sub-Gerente General

Lourdes Carol Arias Ramos
SUB GERENTE GENERAL
SISTEMA INTELIGENTE ERP SAC

www.sistemainteligente.org Jr. Valentín Espejo 448 Oficina 301, San Juan de Miraflores –
Lima, 997014694

Anexo 4. Instrumento de recolección de datos - Ficha de Observación.



FICHA DE OBSERVACIÓN

I. Datos generales

Nombre:

Cargo:

Fecha:

Proyecto:

II. Descripción:

En este documento se registrarán de forma válida y confiable los errores a la interna de los procesos de construcción de software, siguiendo las etapas del modelo de procesos de construcción de software propuesto.

III. Información específica

N°	Fase	Actividad	Defecto encontrado
1			
2			
3			

Anexo 5. Instrumento de recolección de datos - Formulario Google.

CUESTIONARIO

La siguiente encuesta es para ver su satisfacción como cliente, sobre el proyecto desarrollado con nosotros y cuanto nos recomendaría.

1. ¿El software funciona y cumple sus expectativas?

Bueno

Regular

Malo

2. ¿Es la interfaz de nuestro software fácil de usar?

Muy fácil

Fácil

Poco fácil

Difícil

3. ¿Está usted satisfecho con el rendimiento de nuestro software?

Muy satisfecho

Satisfecho

Insatisfecho

4. ¿Cumplimos con la fecha de entrega?

Si

No

5. ¿Te encuentras satisfecho con el producto?

Muy satisfecho

Satisfecho

Insatisfecho

6. Con que nivel recomendarías nuestro software

Muy alto

Alto

Bajo

Anexo 6. Formatos empleados en el desarrollo del modelo

DOCUMENTO	PROPÓSITO/ESTRUCTURA
Plan del Proyecto	<p>Presenta cómo serán ejecutados los procesos y actividades del proyecto para asegurar su conclusión exitosa, así como la calidad de los productos entregables. Puede incluir los siguientes elementos y características:</p> <ul style="list-style-type: none"> • Descripción de producto • Propósito • Requisitos generales del Cliente • Alcance descripción respecto de lo que está incluido y de lo que no está incluido • Objetivos del proyecto • Entregables y/o lista de productos a ser entregados al Cliente • Tareas, incluyendo verificación, validación y revisiones con el Cliente y Equipo de Trabajo que permitan asegurar la calidad de los productos de trabajo. Las Tareas pueden ser representadas como una Estructura de Descomposición del Trabajo (EDT). • Duración estimada de las Tareas • Recursos (humanos, materiales, estándares, equipos y herramientas), incluyendo la capacitación necesaria. Incluye la identificación y programación de los Recursos. • Composición del Equipo de Trabajo. • Calendario de las Tareas del proyecto, indicando la fecha de inicio y fecha de finalización previstas para cada Tarea, y las relaciones y dependencias entre ellas. • Esfuerzo y el costo estimado. • Identificación de los riesgos del proyecto. • Estrategia para el control de versiones. • Herramientas de repositorio del producto o mecanismos identificados.

	<ul style="list-style-type: none"> • Localización y mecanismos de acceso para el repositorio especificado • Identificación y control de versiones definidos. • Respaldo y mecanismos de recuperación definidos • Mecanismos de almacenamiento, manipulación y entrega especificados (incluyendo archivo y recuperación). • Instrucciones de entrega • Elementos requeridos para la liberación del producto (por ejemplo, hardware, software, documentación, etc.) • Requisitos de entrega • Tareas a realizar en orden secuencial • Liberaciones aplicables identificadas • Identifica todos los Componente de software entregados con información de la versión • Identifica cualquier procedimiento de copia de respaldo y recuperación necesarios.
Acta de Reunión	<p>Registro de los acuerdos en reuniones con el cliente y/o el Equipo de Trabajo. Se propone la siguiente estructura:</p> <ul style="list-style-type: none"> • Propósito de la reunión • Asistentes • Fecha y lugar • Referencia a actas de reunión previas • Qué fue logrado • Identifica cuestiones planteadas • Cualquier asunto abierto • Acuerdos • Próxima reunión (en caso necesario)
Acta de Aceptación	<p>Documentación de aceptación por parte del cliente sobre los entregables del proyecto. La estructura propuesta del documento:</p> <ul style="list-style-type: none"> • Registro de la recepción de la entrega

	<ul style="list-style-type: none"> • Identificación de la fecha de recepción • Identificación de los elementos entregados • Registro de la verificación de los criterios de aceptación definidos por parte del cliente • Identificación de cualquier asunto pendiente (en caso de ser aplicable) • Firmado como recibido por parte del cliente.
Solicitud de Cambio	<p>Solicitud de una modificación para corregir un problema o incorporar una mejora en el software o en su documentación. Contiene la siguiente información:</p> <ul style="list-style-type: none"> • Identifica el propósito del cambio • Estado de la solicitud • Información de contacto del solicitante • Sistema(s) impactado(s) • Impacto en la operación de sistemas existentes • Impacto en la documentación asociada <p>Criticidad de la solicitud y fecha en que se requiere</p>
Resultados de Verificación	<p>Documento de la ejecución de la verificación. Incluye el registro de:</p> <ul style="list-style-type: none"> • Participantes • Fecha • Lugar • Duración • Lista de comprobación para la verificación • Elementos aprobados por la verificación • Elementos no aprobados por la verificación • Elementos pendientes de la verificación • Defectos identificados durante la verificación
Registro de Trazabilidad	<p>Documenta la relación entre los requisitos incluidos en la Especificación de Requisitos, los elementos del Diseño de</p>

	<p>software, los Componente de software, los Casos y los Procedimientos de Prueba. Incluye:</p> <ul style="list-style-type: none"> • Especificación de los requisitos por rastrear • Proporciona el mapeo (hacia adelante y hacia atrás) de los requisitos a los elementos del Diseño de software, los Componente de software, los Casos de Prueba y los Procedimientos de Prueba
<p>Manual de Usuario</p>	<p>Describe la forma de uso del Software basado en la interfaz de usuario. Cuenta con las siguientes características:</p> <ul style="list-style-type: none"> • Procedimientos del usuario para realizar tareas específicas utilizando el Software • Procedimientos de instalación y desinstalación • Breve descripción del uso previsto del Software (el concepto de operaciones) • Recursos provistos y requeridos • Entorno operacional requerido • Facilidad para reportar problemas y asistencia • Procedimientos para entrar y salir del Software • Relación y explicación de comandos del Software y de los mensajes del sistema hacia el usuario • Según corresponda a los riesgos identificados; este incluye advertencias, precauciones y notas con correcciones • Incluye los procedimientos para la solución de problemas y corrección de errores • Está escrito en términos comprensibles para los usuarios. Los estados aplicables son: preliminar, verificado e incorporado en la línea base.

<p>Documentación de mantenimiento de software</p>	<p>Cuando se pone marcha un Sistema Informático es necesario que la documentación que explique cómo funciona y qué cometidos específicos tienen cada una de sus partes. Estructura planteada:</p> <ul style="list-style-type: none"> • Descripción Funcional • Manual de Instalación • Manual de Introducción • Manual de Referencia • Guía de Operación o Guía del Operador
<p>Documento de pruebas unitarias</p>	<p>Contiene la lista de pruebas unitarias ejecutadas por el programador. Incluye:</p> <ul style="list-style-type: none"> • Id de la prueba • Nombre de la prueba • Descripción de la prueba • Condiciones de la prueba • Resultado esperado • Resultado obtenido • Fecha de ejecución

Anexo 7. Herramientas empleadas para el Proceso de Gestión de Proyectos

TIPO	HERRAMIENTA	JUSTIFICACIÓN
Herramienta de Gestión	<p>JIRA</p> 	<p>El proceso de Gestión de Proyectos tiende a realizarse de forma descentralizada en Sistema Inteligente ERP SAC, no logrando un adecuado control de las tareas. JIRA permite un control centralizado de la Gestión de Proyectos, así como también ofrece plantillas y colaboración para mejorar la productividad en los proyectos.</p>
Herramienta de Gestión	<p>29110-in-a-box</p> 	<p>Usando el plugin 29110-in-a-box para JIRA, se permite además agregar plantillas y flujos de trabajo basados en ISO 29110, ayudando a la organización a adecuarse fácilmente a este estándar.</p>
Herramienta de Gestión	<p>Trello</p> 	<p>Trello es un software de administración de proyectos con interfaz web y con cliente para iOS y Android para organizar proyectos. Empleando el sistema Kanban, para el registro de actividades con tarjetas virtuales organiza tareas, permite agregar listas, adjuntar archivos, etiquetar eventos, agregar comentarios y compartir tableros. Trello es un tablón virtual en el que se pueden colgar ideas, tareas, imágenes o enlaces. Es versátil y fácil de usar pudiendo usarse para cualquier tipo de tarea que requiera organizar información.</p>

<p>Herramienta de Gestión</p>	<p>MS-Project</p> 	<p>Microsoft Project (o MSP) es un software de administración de proyectos y programas de proyectos desarrollado y comercializado por Microsoft para asistir a administradores de proyectos en el desarrollo de planes, asignación de recursos a tareas, dar seguimiento al progreso, administrar presupuesto y analizar cargas de trabajo.</p> <p>El software Microsoft Office Project en todas sus versiones (la versión 2019 es la más reciente a enero de 2019) es útil para la gestión de proyectos, aplicando procedimientos descritos en el PMBoK del Project Management Institute</p>
-------------------------------	---	---

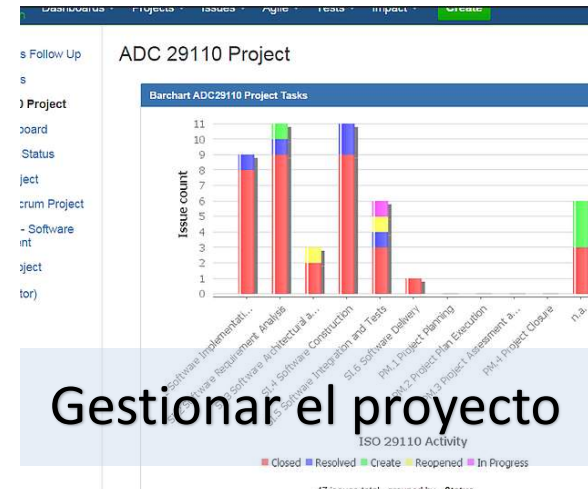
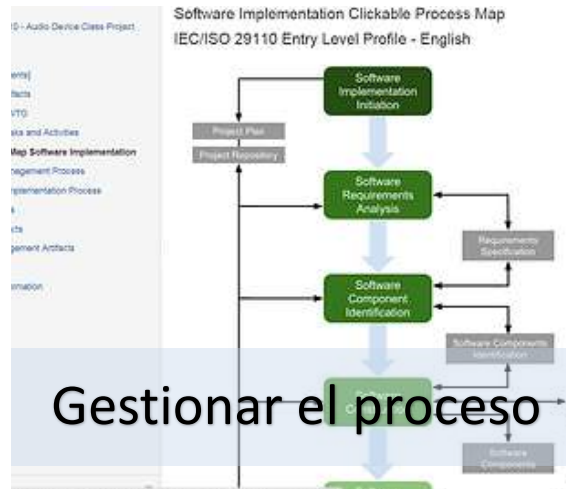
Anexo 8. Herramientas empleadas para el Proceso de Implementación de Proyectos

TIPO	HERRAMIENTA	JUSTIFICACIÓN
Framework de Pruebas Unitarias	JUnit 	Actualmente no se utilizan herramientas automatizadas para la realización de pruebas unitarias. Esta herramienta ayuda a automatizar este proceso reduciendo el tiempo de pruebas.
Herramienta de Cobertura	CodeCover 	La herramienta de cobertura nos permite saber si todo el código fuente es utilizado. Esto evita defectos causados por condiciones que nunca se cumplen en el código.
Herramienta de revisión de código	Findbugs 	Herramienta de revisión de código estático que permite detectar bugs de diversos tipos en Java, así como también, detecta que las buenas prácticas de codificación se estén cumpliendo.
Herramienta para repositorio de proyecto	GitHub 	La herramienta permite alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador.

Anexo 9. Herramienta 29110-in-a-box para el desarrollo de los formatos

Los formatos empleados en el desarrollo del modelo de procesos para la construcción de software de calidad en microempresas peruanas desarrolladoras de software fueron construidos con el soporte de la herramienta 29110-in-a-box, la cual implementa esas guías de una manera simple y directa para permitir administrar y tomar el control del proceso de software en una organización. Cubre la gestión de proyectos y tareas, la gestión de documentos, la gestión de requisitos y pruebas y ayuda a cumplir con la ISO/IEC 29110. 29110-in-a-box se ha instalado en varias empresas: desde startups de 2 personas hasta ingeniería de software de 50 personas. empresa, a un contratista de ingeniería de sistemas de 2.000 grandes.

The screenshot displays a document viewer with a sidebar on the left containing navigation options like 'Home', 'Documents', and 'Projects'. The main content area shows a document titled '2.3 Technical Constraints' and '2.4 Requirements'. The text includes technical specifications for an audio device class project, such as 'The USB Audio Device shall work in pair with a Microsoft HDX host USB' and 'The USB Audio Device shall work in pair with a Microsoft HDX host USB'. A large blue semi-transparent box at the bottom of the screenshot contains the text 'Administrar artefactos'.



Num Sales Follow Up

Mes Tâches

ADC 29110 Project

Gaia Dashboard

Mir Project Status

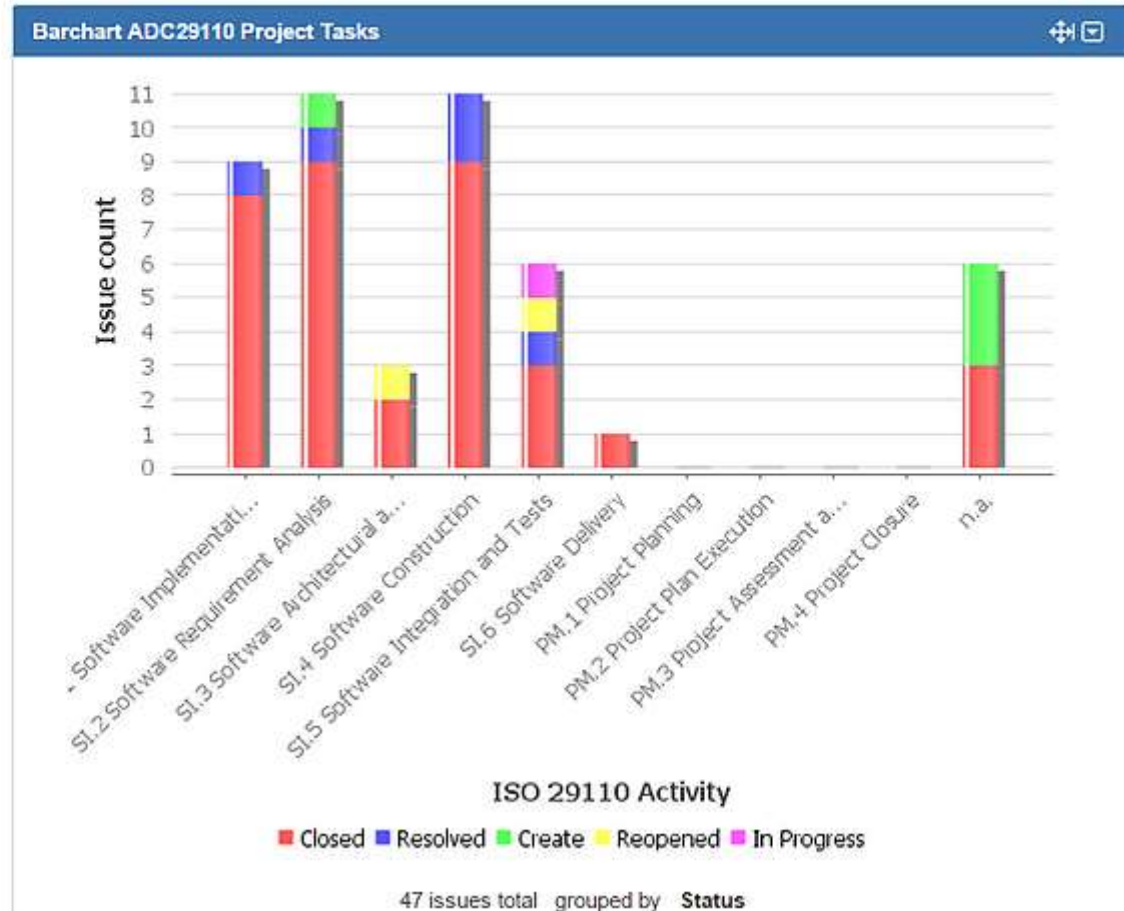
SampleProject

My Agile Scrum Project

[Template] - Software Development

Sample Project (ISO29110 Demonstrator)

ADC 29110 Project



Anexo 10. Acta de constitución de proyecto

	<h2 style="margin: 0;">ACTA DE CONSTITUCIÓN</h2>
---	--

Elaborado por:	Revisado por:	Fecha de primera aprobación:

DATOS DEL SERVICIO						
NOMBRE DEL PROYECTO						
NOMBRE DE LA EMPRESA CLIENTE						
RUC						
TELÉFONO						
DIRECCIÓN						
DATOS REPRESENTANTE LEGAL DE LA EMPRESA						
PROFESIONALES DEL SERVICIO						
PROFESIONAL EN LA PROPUESTA						
ESPECIALIDAD	CARGO	DATOS				OBSERVACIONES
		Nombres y apellidos	N° de Colegiatura	E-mail	Teléfono/celular	

EQUIPO NO INCLUIDO EN LA PROPUESTA					
ESPECIALIDAD	DATOS			COMENTARIO	
	Nombres y apellidos	E-mail	Teléfono/ celular		

ENTREGABLES Y PLAZOS DEL SERVICIO PROGRAMADO (Datos se llenan con las entregas y aprobaciones)					
ENTREGABLES	PLAZO DE ENTREGA	HITO	FECHA DEL HITO	FECHA DE ENTREGA (Según Tiempos Máximos de Entrega)	CÓDIGO
Producto 1					
Producto 2					
Producto 3					
Producto 4					
Producto 5					
MEJORAS INCORPORADAS EN LA PROPUESTA					
CONSIDERACIONES DE PRESENTACIÓN					
Producto 1:					
Producto 2:					
Producto 3:					

Producto 4:							
Producto 5:							
ENTREGABLES Y PLAZOS DEL SERVICIO EJECUTADOS (Completar celdas celeste en ejecución)							
ENTREGAS REALIZADAS	FECHA DE ENTREGA	TIEMPO UTILIZADO	DIAS DE ATRAZO	RESULT. DE LA REV. (APROB U OBSERV.)	FECHA DE RECEP. DE OBSV./APROB.	DIAS DE PLAZO	SEÑALAR EL DOCUMENTO ENTREGADO Y RECIBIDO
PRIMER INFORME							
2DO ENTREGABLE							
OBSER. 2DO ENTREG.							
3ER ENTREGABLE							
OBSER. 3ER ENTREG.							
4TO ENTREGABLE							
5TO ENTREGABLE							
FORMATO DE ENTREGABLES							
TIPO DE LETRA:							
ACTIVIDADES DE CIERRE							
Reunión de cierre de la consultoría							
Encuesta de Satisfacción al Cliente	FECHA DE ENTREGA			FECHA DE RECEPCIÓN		RESULT. DE LA EVALUACIÓN DEL CLIENTE	

Solicitud de prestación de Servicios	FECHA DE ENTREGA		FECHA DE RECEPCIÓN	
HISTORIAL DE CAMBIOS				
FECHA DE ACTUALIZACIÓN	CAMBIO		REALIZADO POR	APROBADO POR

Anexo 11. Formato de Acta de Reunión

ACTA DE REUNIÓN: REUNIONES DE PLANIFICACIÓN				
FECHA	<input style="width: 95%;" type="text"/>	LUGAR	<input style="width: 95%;" type="text"/>	
HORA	<input style="width: 95%;" type="text"/>	HOJA	<input style="width: 95%;" type="text"/>	
ASISTENTES				
NOMBRE Y APELLIDOS				
1		6		
2		7		
3		8		
4		9		
5		10		
AGENDA				
ASUNTO			QUIÉN	
1				
2				
3				
4				
5				
6				
COMPROMISOS				
ASUNTO	QUÉ	QUIÉN	FECHA	
			Estimada	Real
NOTAS				
PRÓXIMA REUNIÓN				
DÍA	<input style="width: 95%;" type="text"/>			
HORA	<input style="width: 95%;" type="text"/>			

Anexo 12. Acta de Aceptación



Formato Acta de Aceptación

Nombre de Proyecto:

Nombre de Cliente:

Declaración de la aceptación formal

Por medio de la presente acta se deja constancia de la finalización y aceptación del proyecto [ID + Nombre de Proyecto] a cargo de [Nombre de Empresa y Nombre de Cliente], iniciado el [día] de [mes] del [año] y culminando el [día] de [mes] del [año].

En este punto se da por concluido el proyecto, por lo que habiendo constatado el SPONSOR, el DUEÑO DE PRODUCTO y el GESTOR DE PROYECTOS la finalización, entrega y aceptación del sistema [Nombre del Software] se certifica la aceptación del proyecto, el cual culmina cumpliendo los requerimientos solicitados.

El proyecto comprendía la entrega de los siguientes entregables:

Iniciación

- Elaboración de contrato para cliente
- Elaboración de Carta de proyecto
- Elaboración del plan de Gestión de la Configuración
- Elaboración del Cronograma de Planeación
- Elaboración de la WBS
- Elaboración Informe de Performance de Trabajo N° 1

Información

- Elaboración de la especificación de Requerimientos
- Elaboración Informe de Performance de Trabajo N° 2

Diseño

Diseño de datos

- Elaboración del Modelo Entidad Relación
- Elaboración del Diccionario de Datos

Diseño arquitectónico

- Elaboración del Diagrama de Componentes

Diseño procedimental

- Elaboración de casos de uso
- Elaboración de Diagramas de Secuencia
- Elaboración de Wireframes
- Elaboración de Propuesta de Diseño de Interfaz
- Elaboración Informe de Performance de Trabajo N° 3

Desarrollo

- Maquetación
- Programación

Anexo 13. Formato de registro de trazabilidad

Matriz de Trazabilidad



Fecha(dd-mm-aaaa): _____

Proyecto: _____

Verificado por: Nombre Fecha(dd-mm-aaaa)

Aprobado por: _____

Numero de Identificación	Texto de la necesidad	Texto del requerimiento	Método de verificación	Título o ID del Caso de Uso	Título o ID del Procedimiento de prueba	Fecha de Verificación	Nombre de la persona que realizó la verificación	Resultado de la verificación

Anexo 14. Matriz de consistencia

TÍTULO	PROBLEMA	OBJETIVOS	HIPÓTESIS	VARIABLES	INDICADORES	TIPO Y DISEÑO DE INVESTIGACIÓN	POBLACIÓN Y MUESTRA
DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC.	¿Cómo mejorar la calidad de los procesos de construcción de software en las microempresas desarrolladoras de software en el Perú?	GENERAL	Con la implementación de un modelo de procesos adaptado se podrá mejorar la calidad de los procesos para la construcción de software en las microempresas desarrolladoras de software en el Perú.	INDEPENDIENTE Modelo de Procesos para la construcción de software	<ul style="list-style-type: none"> • Nivel de Claridad • Nivel de Objetividad • Nivel de Actualidad • Nivel de Organización • Nivel de Suficiencia • Nivel de Intencionalidad • Nivel de Consistencia • Nivel de Coherencia • Nivel de Metodología • Nivel de Pertinencia 	<p style="text-align: center;">Tipo de investigación</p> <p style="text-align: center;">Tecnológica Aplicada</p> <p style="text-align: center;">Diseño de investigación</p> <p style="text-align: center;">Experimental</p>	<p style="text-align: center;">Población</p> <p>La población del presente informe estuvo conformada por cinco (05) modelos para la construcción de software:</p> <ul style="list-style-type: none"> • ISO 29110 • ISO 12207 • CMMI • MOPROSOFT • COMPETISOFT <p style="text-align: center;">Muestra</p> <p>La muestra para este estudio ese delimitó mediante muestreo no probabilístico:</p> <ul style="list-style-type: none"> • ISO 29110 • MOPROSOFT
		ESPECÍFICOS					

Fuente: Elaboración propia

Anexo 15. Ficha para Juicio de Expertos.

VALIDACIÓN DE JUICIO DE EXPERTO INSTRUMENTO DE INVESTIGACIÓN

Título de la investigación:

DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC

Autores:

Bach. Huancas Montenegro Jeiner

Bach. Vargas Moreno Jorge Luis

Objetivo:

El objetivo del presente informe es someter a evaluación el presente modelo de procesos para la construcción de software en microempresas peruanas desarrolladoras de software.

I. DATOS GENERALES DEL EXPERTO

1.1. Apellidos y nombres del experto: _____

1.2. Grado Académico y Profesión: _____

1.3. Áreas de Experiencia Profesional: _____

1.4. Institución donde labora: _____

1.5. Nombre del instrumento motivo de evaluación: **VALIDACIÓN DE EXPERTOS PARA EL MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC.**

II. VALIDACIÓN

Se utilizará los siguientes indicadores y criterios para la evaluación del instrumento: CLARIDAD, OBJETIVIDAD, ACTUALIDAD, ORGANIZACIÓN, SUFICIENCIA, INTENCIONALIDAD, CONSISTENCIA, COHERENCIA, METODOLOGÍA, PERTINENCIA.

Valoración: Deficiente - [50-200], Baja - [250-400], Regular - [450-600], Buena: [650-800], Muy Buena - [850-1000]

Anexo 16. Perfiles profesionales de los expertos evaluadores.

EVALUADOR 01

Mg. JAHAIRA ZULEIKA CAMPOS PEREZ

ORCID: <https://orcid.org/0000-0000-0000-0000>

Graduada en Ingeniería de Sistemas por la Universidad Nacional Pedro Ruiz Gallo - Lambayeque. Maestra en Gerencia en Tecnología de Información y Comunicaciones por la Universidad Privada Antenor Orrego. Candidata a Doctora en Ingeniería de Sistemas e Informática.

Especialista en las áreas de Soporte TI, Proyectos TI, Consultoría en TICs, Marketing digital, Business Intelligence, Oportunidades Digitales para la Exportación.

Ha desempeñado funciones principales dentro del Sector Estatal como en el Privado en las siguientes instituciones:

- Comisión de Promoción del Perú para la Exportación y el Turismo - PROMPERÚ.
- CM Asesores y Consultores SAC.

Amplia experiencia como consultora especializada en implementación de Proyectos de Tecnologías de la Información, con más de diez años de experiencia comprobada en implementación de soluciones a nivel corporativo, habilidad para asimilar rápidamente cualquier plataforma tecnológica, con facilidad de comunicación a gran nivel, con conocimientos del idioma inglés a nivel intermedio.

Ha liderado proyectos de desarrollo de software basadas en la implementación de normas técnicas de calidad.

Actualmente se dedica a la asesoría, capacitación y desarrollo de proyectos de ingeniería del software para diversas instituciones de las localidades de Lambayeque y Lima Metropolitana.

Guía

Aplicativo

REGISTRO NACIONAL DE
**GRADOS ACADÉMICOS Y TÍTULOS
PROFESIONALES**

Resultado

GRUADOCAMPOS PEREZ, JAHAIRA ZULEIKA
DNI 41621631

GRADO O TÍTULOINGENIERA DE SISTEMAS
Fecha de diploma: 03/06/2008
Modalidad de estudios: -

INSTITUCIÓNUNIVERSIDAD NACIONAL PEDRO
RUIZ GALLO
PERU

GRUADOCAMPOS PEREZ, JAHAIRA ZULEIKA
DNI 41621631

GRADO O TÍTULOBACHILLER EN INGENIERIA DE
SISTEMAS
Fecha de diploma: 28/12/2006
Modalidad de estudios: -

INSTITUCIÓNUNIVERSIDAD NACIONAL PEDRO
RUIZ GALLO
PERU

GRUADOCAMPOS PEREZ, JAHAIRA ZULEIKA
DNI 41621631

GRADO O TÍTULOMAESTRA EN GERENCIA EN
TECNOLOGÍA DE INFORMACIÓN Y
COMUNICACIONES

EVALUADOR 02

Ing. Salazar Carranza, Jhonatan David

ORCID: <https://orcid.org/0000-0000-0000-0000>

Graduado en Ingeniería de Sistemas por la Universidad Nacional Pedro Ruiz Gallo - Lambayeque. Actualmente cursando estudios de Maestría en Ingeniería de Sistemas con mención en Dirección de Sistemas y Tecnologías de la Información.

Gerente de EMET SOFT CONSULTING S.A.C – Chiclayo, desde el 2016.
Desarrollador de Sistemas Informáticos, experto en Laravel.

Ha desempeñado funciones principales dentro del Sector Estatal como Especialista en Redes y Comunicaciones, Programador, Soporte TI, Proyectos TI, Consultoría en TICs en las siguientes instituciones públicas:

- Instituto Nacional de Estadística e Informática - INEI.
- Oficina Nacional de Procesos Electorales - ONPE.
- Superintendencia Nacional de Aduanas y de Administración Tributaria - SUNAT.
- EMET SOFT CONSULTING S.A.C.
- Superintendencia Nacional de los Registros Públicos - SUNARP
- UGEL BAGUA.

Amplia experiencia como consultor especializado en implementación de Proyectos de Tecnologías de la Información, con más de once años de experiencia comprobada en implementación de soluciones a nivel corporativo, habilidad para asimilar rápidamente cualquier plataforma tecnológica, con facilidad de comunicación a gran nivel, con conocimientos del idioma inglés a nivel intermedio.

Ha liderado proyectos de implementación de Infraestructura de Tecnologías de la Información en las Instituciones Públicas donde ha laborado, ejecutando proyectos por más de Cuatro Millones de Soles. Con amplia experiencia en procesos de adquisición de equipamiento de T.I (Bienes y Servicios).

REGISTRO NACIONAL DE

GRADOS ACADÉMICOS Y TÍTULOS PROFESIONALES

(**)Si existe alguna observación en tu nombre o DNI [haz clic aquí](#).

Resultado

GRADUADOSALAZAR CARRANZA, JHONATAN DAVID
DNI 43131621

GRADO O TÍTULOBACHILLER EN INGENIERIA DE SISTEMAS
Fecha de diploma: 19/11/2007
Modalidad de estudios: -

INSTITUCIÓNUNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
PERU

GRADUADOSALAZAR CARRANZA, JHONATAN DAVID
DNI 43131621

GRADO O TÍTULOINGENIERO DE SISTEMAS
Fecha de diploma: 29/08/17
Modalidad de estudios: PRESENCIAL

INSTITUCIÓNUNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
PERU

EVALUADOR 03

Mg. (c) JORGE LUIS ECHEGARAY MEDINA

ORCID: <https://orcid.org/0000-0000-0000-0000>

Graduado en Ingeniería de Sistemas y Cómputo por la Universidad Inca Garcilaso de La Vega Asociación Civil. Actualmente cursando estudios de Maestría en Ingeniería de Sistemas con mención en Sistemas de Información en la ciudad de Lima.

Especialista en desarrollo de software a medida para diversas empresas de los diferentes ámbitos empresariales a nivel nacional, situación que le permitió la creación de su empresa propia: CompuRaptor S.A.C. en la cual se desempeña como Gerente General.

Ha desempeñado funciones principales dentro del Sector Estatal como en el Privado en las siguientes instituciones:

- Instituto Nacional de Estadística e Informática - INEI.
- CompuRaptor SAC.
- Transportes Sol de Oro SAC.
- Realiza Negocios e Inversiones SRL.
- Cuerpo General de Bomberos Voluntarios del Perú.

Amplia experiencia como Jefe de Operaciones, Jefe de TI, Analista de Sistemas, Administración de empresas y Desarrollo de Software bajo normas internacionales, con más de diez años de experiencia comprobada en implementación de soluciones a nivel corporativo, habilidad para asimilar rápidamente cualquier plataforma tecnológica, con facilidad de comunicación a gran nivel, con conocimientos del idioma inglés a nivel intermedio.

Ha liderado proyectos de desarrollo de software basadas en la implementación de normas técnicas de calidad.

REGISTRO NACIONAL DE
**GRADOS ACADÉMICOS Y TÍTULOS
PROFESIONALES**

Resultado

GRADUADO ECHEGARAY MEDINA, JORGE LUIS
DNI 10163522

GRADO O TÍTULO BACHILLER EN INGENIERIA DE
SISTEMAS Y COMPUTO
Fecha de diploma: 22/02/2006
Modalidad de estudios: -

INSTITUCIÓN UNIVERSIDAD INCA GARCILASO DE
LA VEGA ASOCIACIÓN CIVIL
PERU

GRADUADO ECHEGARAY MEDINA, JORGE LUIS
DNI 10163522

GRADO O TÍTULO INGENIERO DE SISTEMAS Y
COMPUTO
Fecha de diploma: 17/11/2006
Modalidad de estudios: -

INSTITUCIÓN UNIVERSIDAD INCA GARCILASO DE
LA VEGA ASOCIACIÓN CIVIL
PERU

Anexo 17. Comunicación con los expertos de evaluación.

EVALUADOR 01

Solicito apoyo para evaluación de metodología para Tesis

JEINER HUANCAS MONTENEGRO ·hmontenegrojein@crece.usp.edu.pe
para campos.cmasesores

11:50 (hace 5 horas)

Ing. Jahaira Campos, buen día.

Tenemos el agrado de dirigirnos a usted, le escriben los alumnos Jeiner Huancas Montenegro y Jorge Luis Vargas Moreno, ambos estudiantes de la Escuela Académico Profesional de Ingeniería de Sistemas de la Universidad Señor de Sipán.

Mediante el presente pedirle su valioso apoyo para una evaluación de la metodología de nuestro proyecto de Tesis [Título: DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, que proponemos en nuestra tesis.

La evaluación consta en responder un cuestionario de 10 preguntas. La metodología aplicada en el modelo a evaluar se encuentra basada en la Norma ISO/IEC 29110 y el Modelo MoProSoft, utilizando como lenguaje de modelado ArchiMate para modelar dichas actividades del modelo propuesto.

Esperamos su valioso apoyo, adjuntamos el diseño de la metodología del DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, archivo en formato Word y el documento de validación de expertos que, esperamos nos pueda validar.

Muchas gracias, de antemano, por su tiempo y apoyo brindados.

Saludos cordiales,

Jeiner Huancas Montenegro
Jorge Luis Vargas Moreno

Jeiner Huancas Montenegro
Cel. 928000743
HMONTEGROJEIN@crece.usp.edu.pe

2 archivos adjuntos

Validation juicio de...
HUANCAS_MONTE...

Jahaira Zuleika Campos Pérez
para mí

17:07 (hace 1 hora)

Buenas tardes estimado Jeiner,

Adjunto validación de Juicio.

Un abrazote

CM ASESORES Y CONSULTORES SAC
Jahaira Z. Campos Pérez
Ing. Sistemas CIP
Teléf. 074-227774 / Anexo: 27

Validation juicio de...

EVALUADOR 02

Solicito apoyo para evaluación de metodología. Recibidos x

 **JEINER HUANCAS MONTENEGRO** <jmontenegrojein@crece.ucs.edu.pe>
para jhonatandavid

dom, 4 dic 2020, 10:30

Ing. Jhonatan David Salazar Carranza, buenas noches.

Tenemos el agrado de dirigimos a usted, le escriben los alumnos Jeiner Huancas Montenegro y Jorge Luis Vargas Moreno, ambos estudiantes de la Escuela Académico Profesional de Ingeniería de Sistemas de la Universidad Señor de Sipán.

Mediante el presente pedirle su valioso apoyo para una evaluación de la metodología de nuestro DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, que proponemos en nuestra tesis.

La evaluación consta en responder un cuestionario de 10 preguntas. La metodología aplicada en el modelo a evaluar se encuentra basada en la Norma ISO/IEC 29110 y el Modelo MoProSoft, utilizando como lenguaje de modelado ArchiMate para modelar dichas actividades del modelo propuesto.

Esperamos su valioso apoyo, adjuntamos el diseño de la metodología del DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, archivo en formato Word y el documento de validación de expertos que, esperamos nos pueda validar.


Muchas gracias, de antemano, por su tiempo y apoyo brindados.

Saludos cordiales,
Jeiner Huancas Montenegro
Jorge Luis Vargas Moreno


Jeiner Huancas Montenegro
Cel. 929000743
HMONTENEGROJEIN@crece.ucs.edu.pe



Solicito apoyo para evaluación de metodología. Recibidos x

 **JEINER HUANCAS MONTENEGRO**
Ing. Jhonatan David Salazar Carranza: buenas noches. Tenemos el agrado de dirigimos a usted, le escriben los alumnos Jeiner Huancas Montenegro y Jorge Luis Var

dom, 4 dic 2020, 10:30

 **Jhonatan David Salazar Carranza** <jhonatandavid@gmail.com>
para mí

jue, 10 dic 2020, 15:54

Estimados Jeiner y Jorge,

Adjunto documento con las respuestas de la 1ra iteración.

Atentamente,
Ing. Jhonatan David Salazar Carranza

Responsable de Informática
UGEL Bagua



EVALUADOR 03

Solicito apoyo para evaluación de modelo »

JORGE LUIS VARGAS MORENO <vmorenojorgelui@crece.uss.edu.pe>
para Jorge Echegaray

dom, 6 dic 2020 12:51

Estimado Ing. Jorge Luis Echegaray Medina

Tenemos el agrado de dirigirnos a usted, le escriben los alumnos Jeiner Huancas Montenegro y Jorge Luis Vargas Moreno, ambos estudiantes de Ingeniería de Sistemas de la Universidad Señor de Sipán.

Mediante el presente pedirle su valioso apoyo para una evaluación de la metodología de nuestro DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, que proponemos en nuestra tesis.

La evaluación consta en responder un cuestionario de 10 preguntas. La metodología aplicada en el modelo a evaluar se encuentra basada en la Norma ISO/IEC 29110 y el Modelo MoProSoft, utilizando como lenguaje de modelado ArchiMate para modelar dichas actividades del modelo propuesto.

Esperamos su valioso apoyo, adjuntamos el diseño de la metodología del DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, archivo en formato Word y el documento de validación de expertos que, esperamos nos pueda validar.

Muchas gracias, de antemano, por su tiempo y apoyo brindados.

Saludos cordiales,

Jorge Luis Vargas Moreno
Jeiner Huancas Montenegro

2 archivos adjuntos

Validación juicio de...
Huancas_Vargas_I...

Re: Solicito apoyo para evaluación de modelo » Recibidos

Jorge Luis Echegaray Medina <jorge.echegaray@sistemainteligente.org>
para mí

lun, 7 dic 2020 12:21

Estimado Jorge Vargas

Adjunto lo solicitado, espero haber cumplido con las expectativas.

El dom, 6 dic 2020 a las 12:52, JORGE LUIS VARGAS MORENO (<vmorenojorgelui@crece.uss.edu.pe>) escribió:
Estimado Ing. Jorge Luis Echegaray Medina

Tenemos el agrado de dirigirnos a usted, le escriben los alumnos Jeiner Huancas Montenegro y Jorge Luis Vargas Moreno, ambos estudiantes de Ingeniería de Sistemas de la Universidad Señor de Sipán.

Mediante el presente pedirle su valioso apoyo para una evaluación de la metodología de nuestro DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, que proponemos en nuestra tesis.

La evaluación consta en responder un cuestionario de 10 preguntas. La metodología aplicada en el modelo a evaluar se encuentra basada en la Norma ISO/IEC 29110 y el Modelo MoProSoft, utilizando como lenguaje de modelado ArchiMate para modelar dichas actividades del modelo propuesto.

Esperamos su valioso apoyo, adjuntamos el diseño de la metodología del DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC, archivo en formato Word y el documento de validación de expertos que, esperamos nos pueda validar.

Muchas gracias, de antemano, por su tiempo y apoyo brindados.

Saludos cordiales,

Jorge Luis Vargas Moreno
Jeiner Huancas Montenegro

Validación juicio de...

Anexo 18. Juicio de Expertos.

EVALUADOR 01

VALIDACIÓN DE JUICIO DE EXPERTO INSTRUMENTO DE INVESTIGACIÓN

Título de la investigación:

DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC

Autores:

Bach. Huancas Montenegro Jeiner

Bach. Vargas Moreno Jorge Luis

Objetivo:

El objetivo del presente informe es someter a evaluación el presente modelo de procesos para la construcción de software en microempresas peruanas desarrolladoras de software.

I DATOS GENERALES DEL EXPERTO

- 1.1. Apellidos y nombres del experto: Campos Pérez Jahaira Zuleika
- 1.2. Grado Académico y Profesión: Magíster/Ing. de Sistemas
- 1.3. Áreas de Experiencia Profesional: Soporte TI, Proyectos de TI, Consultoría en TICs, Marketing Digital, Business Intelligence, Oportunidades digitales para la exportación.
- 1.4. Institución donde labora: **PROMPERÚ**
- 1.5. Nombre del instrumento motivo de evaluación: **VALIDACIÓN DE EXPERTOS PARA EL MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC.**

II. VALIDACIÓN

Se utilizará los siguientes indicadores y criterios para la evaluación del instrumento: CLARIDAD, OBJETIVIDAD, ACTUALIDAD, ORGANIZACIÓN, SUFICIENCIA, INTENCIONALIDAD, CONSISTENCIA, COHERENCIA, METODOLOGÍA, PERTINENCIA.

Valoración: Deficiente - [50-200], Baja - [250-400], Regular - [450-600], Buena: [650-800], Muy Buena - [850-1000]

EVALUADOR 02

VALIDACIÓN DE JUICIO DE EXPERTO INSTRUMENTO DE INVESTIGACIÓN

Título de la investigación:

DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC

Autores:

Bach. Huancas Montenegro Jeiner

Bach. Vargas Moreno Jorge Luis

Objetivo:

El objetivo del presente informe es someter a evaluación el presente modelo de procesos para la construcción de software en microempresas peruanas desarrolladoras de software.

I. DATOS GENERALES DEL EXPERTO

- 1.1. Apellidos y nombres del experto: SALAZAR CARRANZA JHONATAN DAVID.
- 1.2. Grado Académico y Profesión: INGENIERO DE SISTEMAS
- 1.3. Áreas de Experiencia Profesional: Especialista en Redes y Comunicaciones, Programador, Soporte TI, Proyectos TI, Consultoría en TICs.
- 1.4. Institución donde labora: UGEL BAGUA
- 1.5. Nombre del instrumento motivo de evaluación: **VALIDACIÓN DE EXPERTOS PARA EL MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC.**

II. VALIDACIÓN

Se utilizará los siguientes indicadores y criterios para la evaluación del instrumento: CLARIDAD, OBJETIVIDAD, ACTUALIDAD, ORGANIZACIÓN, SUFICIENCIA, INTENCIONALIDAD, CONSISTENCIA, COHERENCIA, METODOLOGÍA, PERTINENCIA.

Valoración: Deficiente - [50-200], Baja - [250-400], Regular - [450-600], Buena: [650-800], Muy Buena - [850-1000]

ASPECTOS DE VALIDACIÓN

Indicadores	Criterios	Escala de Evaluación																			
		Deficiente					Baja					Regular					Buena		Muy buena		
		5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
CLARIDAD	Esta formulado con lenguaje apropiado																				X
OBJETIVIDAD	Está expresado en conductas observables																				X
ACTUALIDAD	Adecuado al avance de la gestión del desarrollo de software																				X
ORGANIZACIÓN	Existe una organización lógica																				X
SUFICIENCIA	Comprende los aspectos en cantidad y calidad																				X
INTENCIONALIDAD	Adecuado para valorar la gestión del desarrollo de software																				X
CONSISTENCIA	Basado en aspectos teóricos científicos																				X
COHERENCIA	Entre cada uno de los pasos del modelo																				X
METODOLOGÍA	La estrategia responde al propósito de la investigación																				X
PERTINENCIA	Es útil y adecuado para la investigación																				X

VALORACIÓN: 950

OPINIÓN DE APLICABILIDAD: MUY BUENA PROPUESTA

Lugar y fecha: Chiclayo, 21 de septiembre del 2021.



 MINISTERIO DE EDUCACIÓN
 GOBIERNO REGIONAL AMAZONAS
 UGEL BAJA U.F. 303
 Ing. Ithonán Salazar Carranza
 ESP DE INFORMÁTICA
 N° 43131621
 MODELO DE SISTEMA

EVALUADOR 03

VALIDACIÓN DE JUICIO DE EXPERTOS INSTRUMENTO DE INVESTIGACIÓN

Título de la investigación:

DESARROLLO DE UN MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC

Autores:

Bach. Huancas Montenegro Jeiner

Bach. Vargas Moreno Jorge Luis

Objetivo:

El objetivo del presente informe es someter a evaluación el presente modelo de procesos para la construcción de software en microempresas peruanas desarrolladoras de software.

I. DATOS GENERALES DEL EXPERTO

1.1. Apellidos y nombres del experto: ECHEGARAY MEDINA JORGE LUIS

1.2. Grado Académico y Profesión: INGENIERO DE SISTEMAS | GERENTE GENERAL COMPURAPTOR SAC.

1.3. Áreas de Experiencia Profesional: INGENIERÍA DE SISTEMAS – DESARROLLO DE SOFTWARE A MEDIDA

1.4. Institución donde labora: COMPURAPTOR SAC.

1.5. Nombre del instrumento motivo de evaluación: VALIDACIÓN DE EXPERTOS PARA EL MODELO DE PROCESOS PARA LA CONSTRUCCIÓN DE SOFTWARE EN MICROEMPRESAS PERUANAS DESARROLLADORAS DE SOFTWARE. CASO DE ESTUDIO: SISTEMA INTELIGENTE ERP SAC.

II. VALIDACIÓN

Se utilizará los siguientes indicadores y criterios para la evaluación del instrumento: CLARIDAD, OBJETIVIDAD, ACTUALIDAD, ORGANIZACIÓN, SUFICIENCIA, INTENCIONALIDAD, CONSISTENCIA, COHERENCIA, METODOLOGÍA, PERTINENCIA.

Valoración: Deficiente - [50-200], Baja - [250-400], Regular - [450-600], Buena: [650-800], Muy Buena - [850-1000]

ASPECTOS DE VALIDACIÓN

Indicadores	Criterios	Escala de Valoración																			
		Deficiente				Baja				Regular				Buena				Muy buena			
		5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
CLARIDAD	Esta formulado con lenguaje apropiado																				X
OBJETIVIDAD	Está expresado en conductas observables																			X	
ACTUALIDAD	Adecuado al avance de la gestión del desarrollo de software																				X
ORGANIZACIÓN	Existe una organización lógica																		X		
SUFICIENCIA	Comprende los aspectos en cantidad y calidad																				X
INTENCIONALIDAD	Adecuado para valorar la gestión del desarrollo de software																		X		
CONSISTENCIA	Basado en aspectos teóricos científicos																				X
COHERENCIA	Entre cada uno de los pasos del modelo																				X
METODOLOGÍA	La estrategia responde al propósito de la investigación																				X
PERTINENCIA	Es útil y adecuado para la investigación																				X

VALORACION: 935

OPINIÓN DE APLICABILIDAD: MUY BUENA PROPUESTA

Lugar y fecha: Chiclayo, 20 Septiembre del 2021.

Jorge Chirigary

Anexo 19. Diapositivas para capacitación del modelo propuesto

Huancas_&_Vargas_SU - PowerPoint

Inicio Insertar Dibujar Diseño Transiciones Animaciones Presentación con diapositivas Revisar Vista Grabación Ayuda ¿Qué desea hacer? Compartir

Está compartiendo la pantalla Deja de

Método Propuesto

Desarrollo de un modelo de procesos para la construcción de software en microempresas peruanas desarrolladoras de software. Caso de estudio: Sistema Inteligente ERP SAC.

Figura 17. Vista del Proceso de Gestión de Proyectos.
Fuente. Elaboración propia

Slide 14

LABSIS
Laboratorio de Sistemas Inteligentes y Seguridad Informática

Diapositiva 14 de 21 Español (España) 87%

Anexo 20. Capacitación del Modelo en Sistema Inteligente ERP SAC

The screenshot displays a Microsoft Word document titled "HUANCAS, MONTENEGRO, & VARGAS, MORENO.docx". The document content features a central diagram titled "Gestión del Proyecto" (Project Management) highlighted in yellow. The diagram consists of a central horizontal flow of four boxes: "PM01 Planificación del Proyecto", "PM02 Ejecución del Plan del Proyecto", "PM03 Evaluación y Control del Proyecto", and "PM04 Cierre del Proyecto". Surrounding this central flow are seven other boxes, each representing a specific objective: "PM.01 Se desarrolla el plan del proyecto", "PM.02 Se monitorea el progreso del proyecto", "PM.03 Se abordan las solicitudes de cambio", "PM.04 Revisar reuniones", "PM.05 Se desarrolla la estrategia de control de versiones", "PM.05 Se identifican los riesgos", and "PM.07 Se realiza la garantía de calidad del software". Arrows indicate relationships between these objectives and the central process steps.

Below the diagram is the caption: "Figura 37. Objetivos de la vista del Proceso de Gestión de Proyectos." and the source: "Fuente, Elaboración propia".

Overlaid on the right side of the document is a "Participantes (4)" window from a meeting application. It lists the following participants: Jorge Vargas (Anfitrión, yo), JEINER HUANCAS MONTENEGRO, marcos vargas, and Salomon Gabriel. Each participant has icons for mute, video, and chat. At the bottom of the window are buttons for "Invitar" and "Silenciar a todos".

The bottom of the screenshot shows the Windows taskbar with the system tray displaying "18°C Bruma" and the date "27/09/2021".

HUANCAS_MONTENEGRO_q_VARGAS_MORENO.docx - Word

Está compartiendo la pantalla

Figura 39, Objetivos de la vista del Proceso de Implementación del Software

Fuente, Elaboración propia

130

Participantes (4)

- Jorge Vargas (Anfitrión, yo)
- JEINER HUANCAS MONTENEGRO
- marcos vargas
- Salomon Gabriel

Invitar Silenciar a todos

Página 130 de 176 · 32807 palabras · Español (Perú)

18°C Bruma · 27/09/2021 12:00

Anexo 21. Proyectos evaluados Pres Test

Proyecto 01: Sistema de Control de Asistencia (PRE-TEST)

UGEL BAGUA Inicio Nosotros Cerrar Sesión

Persona
Usuarios
Tipo de Usuarios
Horarios
Reporte diario de marcaciones
Olvido de marcaciones
Reporte diario de asistencias
Reporte detallado de asistencias
Reporte de inasistencias por periodo

Sistema de Control de Asistencia

Persona

[NUEVO](#) [Descargar Excel](#)

ID	Nombres	DNI	Genero	Fecha Nac.	Correo	Teléfono	Estado	Cargo	Horario	Acciones
1	Jeiner Huancas Montengero	47059330	M	1992-05-12	jhuancas@ugelbagua.gob.pe	928000743	A	Practicante	Horario01	Editar Ver Eliminar
11	Jackson Jesús Fuentes Perez	12345679	M	2015-08-11	ddog@ugelbagua.gob.pe	928000743	A	Coordinador	Horario02	Editar Ver Eliminar
30	Suly Torres Dogs	33214567	F	1987-02-03	storres@ugelbagua.gob.pe	987456321	A	Practicante	Part Time 02	Editar Ver Eliminar

UGEL BAGUA Inicio Nosotros Cerrar Sesión

Persona
Usuarios
Tipo de Usuarios
Horarios
Reporte diario de marcaciones
Olvido de marcaciones
Reporte diario de asistencias
Reporte detallado de asistencias
Reporte de inasistencias por periodo
Informes

Sistema de Control de Asistencia

Reporte Detallado de Asistencias

[Descargar Excel](#)

DNI	Nombres	Dia	Horario	Entrada01	Salida01	Tardanza01	Entrada02	Salida02	Tardanza02	Tardanzas	Estado	Observación
47059330	Huancas Montengero Jeiner	Lunes	Horario01	0	0	05:00:00	0	0	03:00:00	08:00:00	●	ES SU DIA
12345679	Fuentes Perez Jackson Jesús	Lunes	Horario02	0	0	04:00:00	0	0	00:00:00	08:00:00	●	ES SU DIA
33214567	Torres Dogs Suly	Lunes	Part Time 02	0	0	00:00:00	0	0	00:00:00	00:00:00	●	NO ES SU DIA

Proyecto 02: Sistema de detracciones y retenciones (PRE-TEST)

|| Sistema de Tesorería - CONSORCIO DHMONT & CG & M S.A.C. ||
Mantenimientos
Procesos
Reportes
Consultas
Utilitarios
Configuración
Ventana
Ayuda
Salir

Nuevo: F2
Modificar: F1
Eliminar: Sup
Guardar: F5
Cancelar: F8
Borrar: F4
Exportar: F1
Imprimir: F7
Actualizar: F11
Salir: Esc

Reporte Programación de Detracciones

Año: 2019
Mes: S

F. Programación	Número	F. Constancia	Nro Constancia	Nro Pago	Fecha	F. PROG. Pago	Proveedor	Ruc	F. Emisión	Serie	Número	Factura	Imp. Factura	Cta Detra
30/09/2019	0014	17/09/2019	103250220	160895	12/09/2019	12/09/2019	GESERPRO PROYECTOS Y SERVICIOS...	20603380291	31/08/2019	F001	8	F001-8	167423.55	00008023
27/09/2019	0012	17/09/2019	103250221	160895	12/09/2019	12/09/2019	AGREGADOS & MARMOLES DEL PE...	20516157519	4/09/2019	F001	1370	F001-1370	27304.50	00005179
26/09/2019	0011	17/09/2019	103250222	160895	12/09/2019	12/09/2019	AGREGADOS SAINT THOMAS S.A.C.	20550638771	31/08/2019	E001	150	E001-150	916.00	00001095
25/09/2019	0010	17/09/2019	103250223	160895	12/09/2019	12/09/2019	ACUÑA CERVANTES MOISES DAVID	10453356979	12/08/2019	002	000118	002-000118	2517.00	00074124
23/09/2019	0009	17/09/2019	103250224	160895	12/09/2019	12/09/2019	AGREGADOS CARAPONGO E.I.R.L.	20509550957	3/09/2019	E001	322	E001-322	24006.45	00000485
17/09/2019	0007	17/09/2019	103250225	160895	12/09/2019	12/09/2019	QUIMICA MARTELL S.A.C.	20501820025	28/08/2019	FA01	00010...	FA01-0001012	2800.00	00026006
17/09/2019	0008	17/09/2019	103250226	160895	12/09/2019	12/09/2019	AGREGADOS PICA PIEDRA S.R.L.	20602358942	29/08/2019	001	000173	001-000173	13556.43	00028067
14/09/2019	0006	17/09/2019	103250227	160895	12/09/2019	12/09/2019	VALVERDE PEÑA FEDERICO	10028799468	1/08/2019	E001	21	E001-21	8024.00	00631218
12/09/2019	0005	17/09/2019	103250228	160895	12/09/2019	12/09/2019	OPERACIONES DE AGREGADOS DEL...	20602931651	27/08/2019	E001	144	E001-144	6400.00	00028068
10/09/2019	0004	17/09/2019	103250229	160895	12/09/2019	12/09/2019	OPERACIONES DE AGREGADOS DEL...	20602931651	5/09/2019	E001	145	E001-145	760.00	00028069
6/09/2019	0003	17/09/2019	103250230	160895	12/09/2019	12/09/2019	OPERACIONES DE AGREGADOS DEL...	20602931651	5/09/2019	E001	146	E001-146	2200.00	00028066
5/09/2019	0002	17/09/2019	103250231	160895	12/09/2019	12/09/2019	OPERACIONES DE AGREGADOS DEL...	20602931651	5/09/2019	E001	149	E001-149	800.00	00028066
4/09/2019	0001	17/09/2019	103250232	160895	12/09/2019	12/09/2019	OPERACIONES DE AGREGADOS DEL...	20602931651	5/09/2019	E001	150	E001-150	16000.00	00028066
		17/09/2019	103250233	160895	12/09/2019	12/09/2019	DHMONT S.A.C. CONTRATISTAS GE...	20108664640	9/09/2019	F010	5	F010-5	8496.00	00098001
		17/09/2019	103250234	160895	12/09/2019	12/09/2019	MSTEEL ASCENSORES S.A.C.	20538014550	31/08/2019	F001	47	F001-47	120153.07	00005205
		17/09/2019	103250235	160895	12/09/2019	12/09/2019	JL VITTERI INGENIEROS SOCIEDAD A...	20509882950	2/09/2019	F002	00032...	F002-0003268	6372.00	00000526
		17/09/2019	103250236	160895	12/09/2019	12/09/2019	JL VITTERI INGENIEROS SOCIEDAD A...	20509882950	2/09/2019	F002	00032...	F002-0003269	6372.00	00000526

Grabar Constancia de Pago
Generar Detracción

Registro de Retenciones

Año: 2019 Mes: Septiembre Todo el año 21,390.09 Total Retención S/.

Imprimir Txt Resumen Di... Generar Txt P... Excel

Drag a column here to group by this column.

Fecha Pago	Serie	Nro Retención	Serie Et...	Numer...	Proveedor	Ruc	Total Retención S/.	Documento	Reservado?	Anulado?	Idasient...
30/09/2019	R010	422	R010	422	MERCUR PERU E.I.R.L.	20602703780	122.22	FACTURA DE COMPRA	NO	NO	949467
30/09/2019	R010	423	R010	423	MECHANICAL WORLD S.A.C.	20506591431	35.45	FACTURA DE COMPRA	NO	NO	1145220
30/09/2019	R010	424	R010	424	MECHANICAL WORLD S.A.C.	20506591431	33.93	FACTURA DE COMPRA	NO	NO	949470
30/09/2019	R010	425	R010	425	MAVEGSA DRYWALL S.A.C.	20546034951	101.43	FACTURA DE COMPRA	NO	NO	949471
30/09/2019	R010	426	R010	426	SEGURIDAD INTELIGENTE S.A.C.	20602387951	155.40	FACTURA DE COMPRA	NO	NO	949472
30/09/2019	R010	427	R010	427	INVERSIONES LOREN7 S.A.C.	20551048757	90.22	FACTURA DE COMPRA	NO	NO	949473
30/09/2019	R010	428	R010	428	TAYM INVERSIONES & ASESORIA INTEGRAL S.A.C.	20601350417	75.57	FACTURA DE COMPRA	NO	NO	949474
30/09/2019	R010	429	R010	429	CORPORACION M & K PERU S.A.C.	20512175199	27.52	FACTURA DE COMPRA	NO	NO	949475
30/09/2019	R010	430	R010	430	ADITIVOS ESPECIALES S.A.C	20517242161	149.74	FACTURA DE COMPRA	NO	NO	949476
30/09/2019	R010	431	R010	431	FORESTAL SANTA ROSA S.A.C.	20521872145	1,632.09	FACTURA DE COMPRA	NO	NO	1145222
30/09/2019	R010	432	R010	432	FORESTAL SANTA ROSA S.A.C.	20521872145	557.37	FACTURA DE COMPRA	NO	NO	949479
30/09/2019	R010	433	R010	433	NEXUS PC SERVICIOS INFORMATICOS S.A.C.	20600325192	165.40	FACTURA DE COMPRA	NO	NO	949480
30/09/2019	R010	434	R010	434	PLACACENTRO PUENTE PIEDRA E.I.R.L.	20601719739	68.55	FACTURA DE COMPRA	NO	NO	1145223
30/09/2019	R010	435	R010	435	HIDROMECC INGENIEROS SOCIEDAD ANONIMA CERRADA	20514001783	44.07	FACTURA DE COMPRA	NO	NO	949483
30/09/2019	R010	436	R010	436	ALTERA - ACEROS E INVERSIONES SOCIEDAD COMERCIAL DE RES...	20477913599	76.32	FACTURA DE COMPRA	NO	NO	949484
30/09/2019	R010	437	R010	437	COEPEP PERU S.A.C.	20602964320	75.60	FACTURA DE COMPRA	NO	NO	949485
30/09/2019	R010	438	R010	438	T Y T DISTRIBUIDORA FERRETERA S.A.C.	20513956976	42.14	FACTURA DE COMPRA	NO	NO	949486

136 Registros

Proyecto 03: Plataforma para el control de parqueo vehicular (PRE-TEST)

Encuesta Refrescar

Periodo 2021

Introduzca el texto a buscar...

fecha	Nombre	Area	Recomendacion	Observacion
Calificacion: MALA (3)				
Calificacion: REGULAR (1)				

4 Registros

Anexo 22. Proyectos evaluados Post Test

Proyecto 01: Aplicativo móvil de control triaje covid-19 (POST-TEST)

22:50

← COVID-19

Declaración Jurada

Todos los datos expresados en esta ficha constituyen declaración jurada de mi parte. He sido informado que de omitir o falsear información puedo perjudicar la salud de mis compañeros, y la mía propia, lo cual constituir una falta grave a la salud pública, asumo sus consecuencias

Evento

INGRESO

Sitio

OFICINA SAN ISIDRO

Sensación de alza térmica o fiebre 38

En los últimos 14 días calendario he tenido alguno de los síntomas siguientes:

Tos, estornudos o dificultad para respirar

Expectoración o flema amarilla o verdosa

Contacto con persona(s) con un caso confirmado de COVID-19

Está tomando alguna medicación (detallar cuál o cuáles) SI

22:52

← COVID-19

Declaración Jurada

Todos los datos expresados en esta ficha constituyen declaración jurada de mi parte. He sido informado que de omitir o falsear información puedo perjudicar la salud de mis compañeros, y la mía propia, lo cual constituir una falta grave a la salud pública, asumo sus consecuencias

Evento

SALIDA

Sitio

OFICINA SAN ISIDRO

Foto Trabajo Foto Salida

ACEPTAR

22:51

← COVID-19

Sensación de alza térmica o fiebre 38

En los últimos 14 días calendario he tenido alguno de los síntomas siguientes:

Tos, estornudos o dificultad para respirar

Expectoración o flema amarilla o verdosa

Contacto con persona(s) con un caso confirmado de COVID-19

Está tomando alguna medicación (detallar cuál o cuáles) SI

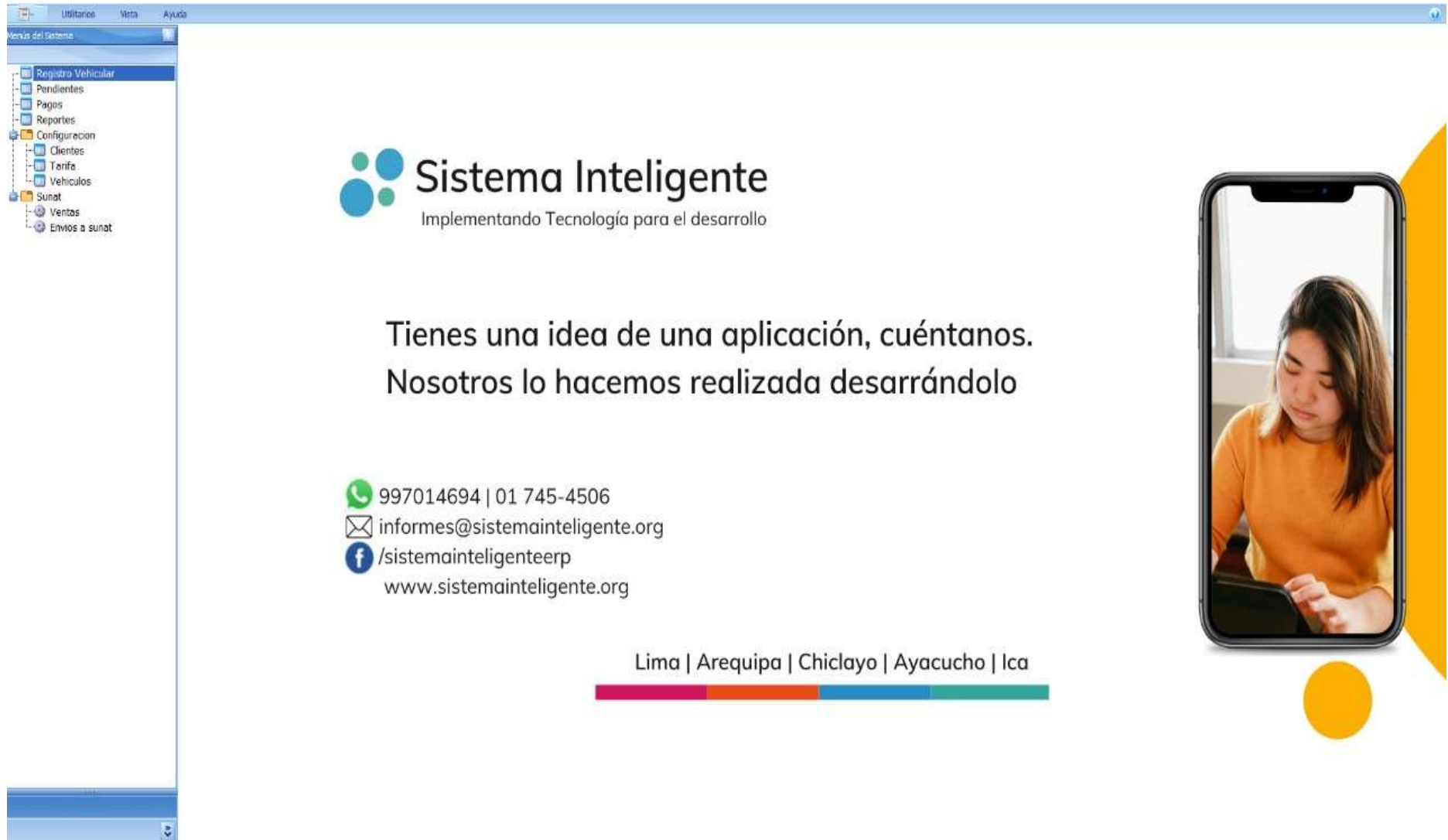
Firma:

Borrar

Firma

ACEPTAR


Proyecto 02: Plataforma de administración de seguridad (POST-TEST)






The image shows a screenshot of a web application interface. On the left, there is a sidebar menu titled 'Menú del Sistema' with the following items: Registro Vehicular, Pendientes, Pagos, Reportes, Configuración, Clientes, Tarifa, Vehículos, Sunat, Ventas, and Envíos a sunat. The main content area features the logo for 'Sistema Inteligente' (three blue circles) and the tagline 'Implementando Tecnología para el desarrollo'. Below this is the text: 'Tienes una idea de una aplicación, cuéntanos. Nosotros lo hacemos realizada desarrollándolo'. Contact information is provided: a WhatsApp icon with the number 997014694 | 01 745-4506, an email icon with inform@systemainteligente.org, a Facebook icon with /systemainteligenteerp, and the website www.systemainteligente.org. At the bottom, a horizontal bar lists the cities: Lima | Arequipa | Chiclayo | Ayacucho | Ica. To the right of the main content, there is a vertical yellow bar and a smartphone displaying a woman in an orange top looking at a tablet. A yellow circle is positioned below the smartphone.

Menú del Sistema


- Registro Vehicular
- Pendientes
- Pagos
- Reportes
- Configuración
- Clientes
- Tarifa
- Vehículos
- Sunat
- Ventas
- Envíos a sunat

 **Sistema Inteligente**
Implementando Tecnología para el desarrollo

Tienes una idea de una aplicación, cuéntanos.
Nosotros lo hacemos realizada desarrollándolo

 997014694 | 01 745-4506
 inform@systemainteligente.org
 /systemainteligenteerp
www.systemainteligente.org

Lima | Arequipa | Chiclayo | Ayacucho | Ica



Proyecto 03: Control total de seguridad desde tu smartphone (POST-TEST)



Anexo 23. Cronogramas en MS-Project de proyectos desarrollados

APLICATIVO MOVIL CONTROL DE ASITENCIA																						
TÍTULO DEL PROYECTO		APLICATIVO MOVIL BOXER CONTROL																				
RESPONSABLE DEL PROYECTO		JORGE LUIS VARGAS MORENO																				
NÚMERO EDT	TÍTULO DE LA TAREA	RESPONSABLE DE LA TAREA	FECHA DE INICIO	FECHA DE ENTREGA	DURACIÓN	% COMPLETADO DE LA TAREA	FASE UNO		FASE DOS		FASE TRES				FASE 4							
							SEMANA	SEMANA	SEMANA	SEMANA	SEMANA	SEMANA										
							1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
1 Fase Uno - Análisis e inicio del proyecto																						
1.1	Firma de Acta del proyecto	Jorge Vargas	13/04/20	17/04/20	4	100 %	x															
1.1.1	Recepción y análisis del Requerimiento de la empresa	Jorge Vargas	20/04/20	24/04/20	4	100 %		x														
1.2	Reunión con los Usuarios Cliente	Jorge Vargas	27/04/20	01/05/20	4	100 %			x													
2 Fase Dos - Definición y planificación del proyecto																						
2.1	Elaboracion de Prototipo	Jorge Vargas	04/05/20	08/05/20	4	100 %				x												
2.2	Reunión con el cliente aprobando las funcionalidad del producto	Jorge Vargas	11/05/20	15/05/20	4	100 %					x											
3 Fase Tres - Desarrollo del proyecto																						
3.1	Elaboración de diagrama ER Sprint Daily Meeting - reunión diaria	Jorge Vargas	18/05/20	22/05/20	4	100 %						x										
3.2	Elaboración de la Base de Datos en SQL Server 2012 o superior. Sprint Daily Meeting - reunión diaria	Jorge Vargas	25/05/20	05/06/20	10	100 %							x									
3.2.1	Desarrollo de Aplicación Movil por módulos. Sprint Daily Meeting - reunión diaria.	Jorge Vargas	08/06/20	03/07/20	25	100 %								x	x	x	x					
3.2.2	Desarrollo de Aplicación Web por módulos. Sprint Daily Meeting - reunión diaria.	Jorge Vargas	06/07/20	24/07/20	18	100 %													x	x	x	
4 Fase Cuatro - Implementación y Entrega del proyecto																						
4.1	Implementación en el cliente del aplicativo Boxer Control y la plataforma web de administracion	Jorge Vargas	27/07/20	28/07/20	1	100 %															x	
4.2	Ejecutable para su distribución en las plataformas de aplicaciones móviles	Jorge Vargas	29/07/20	30/07/20	1	100 %																x
4.3	Entrega del Proyecto	Jorge Vargas	31/07/20	31/07/20	0	0 %																x

APLICATIVO MOVIL TRACK ALARMAS

TÍTULO DEL PROYECTO APLICATIVO MOVIL FICHA TRIAJE COVID - 19

RESPONSABLE DEL PROYECTO JORGE LUIS VARGAS MORENO

NÚMERO EDT	TÍTULO DE LA TAREA	RESPONSABLE DE LA TAREA	FECHA DE INICIO	FECHA DE ENTREGA	DURACIÓN	% COMPLETADO DE LA TAREA	Sprint 01		Sprint 02		Sprint 03				Sprint 04		
							SEMANA			SEMANA		SEMANA				SEMANA	
							1	2	3	4	5	6	7	8	9	10	11
1 Sprint 01 - Análisis e inicio del proyecto																	
1.1	Firma de Acta del proyecto	Jorge Vargas	07/09/20	11/09/20	4	100 %	X										
1.1.1	Recepción y análisis del Requerimiento de la empresa	Jorge Vargas	14/09/20	18/09/20	4	100 %		X									
1.2	Reunión con los Usuarios Cliente	Jorge Vargas	21/09/20	25/09/20	4	100 %			X								
2 Sprint 02 - Definición y planificación del proyecto																	
2.1	Elaboracion de Prototipo	Jorge Vargas	28/09/20	02/10/20	4	100 %			X								
2.2	Reunión con el cliente aprobando las funcionalidad del producto	Jorge Vargas	05/10/20	09/10/20	4	100 %				X							
3 Sprint 03 - Desarrollo del proyecto																	
3.1	Elaboración de diagrama ER Sprint Daily Meeting - reunión diaria	Jorge Vargas	11/10/20	15/10/20	4	100 %				X							
3.2	Elaboración de la Base de Datos en SQL Server 2012 o superior. Sprint Daily Meeting - reunión diaria	Jorge Vargas	17/10/20	27/10/20	10	100 %					X						
3.2.1	Desarrollo de Aplicación Movil por módulos. Sprint Daily Meeting - reunión diaria.	Jorge Vargas	30/10/20	25/11/20	25	100 %					X	X	X	X			
3.2.2	Desarrollo de Aplicación Web por módulos. Sprint Daily Meeting - reunión diaria.	Jorge Vargas	27/11/20	15/12/20	18	100 %							X	X	X		
4 Sprint 04 - Implementación y Entrega del proyecto																	
4.1	Implementación en el cliente del aplicativo Boxer Control y la plataforma web de administracion	Jorge Vargas	16/12/20	17/12/20	1	100 %									X		
4.2	Ejecutable para su distribución en las plataformas de aplicaciones móviles	Jorge Vargas	18/12/20	19/12/20	1	100 %										X	
4.3	Entearea del Provento	Jorae Vargas	20/12/20	20/12/20	0	100 %											X

Anexo 24. Evidencias fotográficas



Fuente, elaboración propia.



Fuente, elaboración propia.