

SOFTWARE PARA LA GESTIÓN DE REQUISITOS EN LOS NIVELES 1 Y 2 DE LA
NORMA ISO/IEC 15504 (SPICE) EN LA EMPRESA SOLINCES S.A.S

Investigadores:

Ing. PLINIO PUELLO MARRUGO

Co-investigadores:

CARLOS YESID HERNÁNDEZ HERRERA

LEIMAN ALFONSO MEJÍA MAURY



UNIVERSIDAD DE CARTAGENA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
CARTAGENA DE INDIAS

2014

SOFTWARE PARA LA GESTIÓN DE REQUISITOS EN LOS NIVELES 1 Y 2 DE LA NORMA
ISO/IEC 15504 (SPICE) EN LA EMPRESA SOLINCES S.A.S

GRUPO DE INVESTIGACIÓN: E-SOLUCIONES
LÍNEA DE INVESTIGACIÓN: E SERVICIOS

Investigadores:

Ing. PLINIO PUELLO MARRUGO

Co-investigadores:

CARLOS YESID HERNÁNDEZ HERRERA

LEIMAN ALFONSO MEJÍA MAURY



UNIVERSIDAD DE CARTAGENA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
CARTAGENA DE INDIAS

2014

DEDICATORIA

Quiero agradecer a Dios, por brindarme el don de la vida y la sabiduría, darme salud y las fuerzas necesarias a lo largo de todo mi trayecto.

A mi madre Amparo Herrera Palmet, que con su apoyo, fortaleza y amor en toda la fase de mi vida me permitió lograr este objetivo tan importante para mí, sé que en este mundo no hay nadie que me ame más que ella, y siempre lo tendré presente.

A Samuel Viana, a mis hermanas Keyla Tatiana y Alix Julieth por su apoyo moral y a cada uno de mis familiares que hicieron posible mi permanencia.

A la Universidad de Cartagena, por abrirme sus puertas y brindarme la oportunidad de cursar mis estudios superiores. A todos los profesores que me brindaron parte de sus conocimientos durante esta etapa ayudando a formarme como profesional.

A mi tutor Plinio Puello Marrugo, por recibirme como su asesorado y por brindarme su apoyo, conocimiento y paciencia a lo largo de todos estos meses, con la finalidad de concluir un buen trabajo. Gracias profesor por su colaboración.

Carlos Yesid Hernández Herrera

CONTENIDO

	Pág.
RESUMEN.....	10
ABSTRACT	12
INTRODUCCIÓN.....	14
1. PLANTEAMIENTO DEL PROBLEMA	16
1.1. DESCRIPCIÓN DEL PROBLEMA.....	16
1.2. FORMULACIÓN DEL PROBLEMA.....	19
2. JUSTIFICACIÓN DEL PROBLEMA	20
3. MARCO DE REFERENCIA	22
3.1. ESTADO DEL ARTE	22
3.1.1. HERRAMIENTAS EXISTENTES PARA LA GESTIÓN DE PROYECTOS SOFTWARE	23
3.1.2. LISTADO DE EMPRESAS QUE APLICAN EL MODELO SPICE.....	25
3.1.3. LISTADO DE EMPRESAS QUE APLICAN EL MODELO CMMI	26
3.2. MARCO TEÓRICO	29
3.2.1. GESTIÓN DE PROYECTOS	29
3.2.2. CALIDAD DEL SOFTWARE.....	31
3.2.3. METODOLOGÍAS DE DESARROLLO.....	31
3.2.4. MODELOS DE PROCESOS DE SOFTWARE	32
3.2.5. MODELOS DE EVALUACIÓN DE PROCESOS.....	33
2.2.5. NIVELES DE ORGANIZACIÓN	49
3.3. ANTECEDENTES	51
4. OBJETIVOS.....	53
4.1. OBJETIVO GENERAL.....	53
4.2. OBJETIVOS ESPECÍFICOS	53
5. METODOLOGÍA	54
5.1. TIPO DE INVESTIGACIÓN	54
5.2. FASES METODOLOGÍA RUP	55
5.2.1. FASE DE INICIO.....	55
5.2.2. FASE DE ELABORACIÓN	59

5.2.3.	FASE DE CONSTRUCCIÓN	60
5.2.4.	FASE DE TRANSICIÓN.....	60
6.	RESULTADOS Y DISCUSIÓN.....	62
6.1.	RESULTADOS DEL OBJETIVO 1.....	62
6.2.	RESULTADOS DEL OBJETIVO 2.....	68
6.3.	RESULTADOS DEL OBJETIVO 3.....	69
	Descripción Caso de Uso	71
6.4.	RESULTADOS DEL OBJETIVO 4.....	83
	CONCLUSIONES Y RECOMENDACIONES	89
	BIBLIOGRAFÍA	93

TABLA DE ILUSTRACIONES

	Pág.
Ilustración 1: Niveles de Madurez de SPICE	34
Ilustración 2: Desarrollo de requisitos.....	44
Ilustración 3: Niveles de Madurez por CMMI	48
Ilustración 4: Estructura de la metodología RUP	54
Ilustración 5: Análisis de Planeación del Proyecto. Fuente (Autores)	64
Ilustración 6: Análisis de Suministro. Fuente (Autores)	65
Ilustración 7: Análisis Definición de los Requisitos de Usuarios. Fuente (Autores)	66
Ilustración 8: Análisis de los Requisitos del Sistema.....	67
Ilustración 9: Caso de Uso General	70
Ilustración 10: Encuesta de usabilidad realizada al representante legal. Fuente (Autores)..	85
Ilustración 11: Encuesta de usabilidad realizada a la ingeniera jefe de proyectos. Fuente (Autores).....	86
Ilustración 12: Encuesta de usabilidad realizada a los analistas. Fuente (Autores)	87

ÍNDICE DE TABLAS

	Pág.
Tabla 1: Herramientas WEB	23
Tabla 2: Herramientas de Escritorio.....	24
Tabla 3: Herramientas de Código Abierto.....	24
Tabla 4: Empresas certificadas por el modelo SPICE.....	25
Tabla 5: Empresas certificadas por CMMI en Latinoamérica.....	26
Tabla 6: Empresas Certificadas por CMMI en Colombia	27
Tabla 7: Dimensión del proceso	37
Tabla 8: Tipos de requisitos	43
Tabla 9: Planeación del Proyecto	57
Tabla 10: Porcentaje de Cumplimiento basado en la Norma ISO/IEC 15504	58
Tabla 11: Modelos de Software.....	68
Tabla 12: Caso de Uso – Agregar Organización	71
Tabla 13: Caso de Uso - Editar Organización	71
Tabla 14: Caso de Uso - Inactivar Organización.....	72
Tabla 15: Caso de Uso - Asignar Usuario	72
Tabla 16: Caso de Uso - Agregar Usuario.....	73
Tabla 17: Caso de Uso - Editar Usuario	73
Tabla 18: Caso de Uso - Inactivar Usuario.....	74
Tabla 19: Caso de Uso - Crear Proyecto	74
Tabla 20: Caso de Uso - Editar Proyecto	75
Tabla 21: Caso de Uso - Eliminar Proyecto	75
Tabla 22: Caso de Uso - Asignar Analista	76
Tabla 23: Caso de Uso - Crear Necesidad.....	76
Tabla 24: Caso de Uso - Editar Necesidad.....	77
Tabla 25: Caso de Uso - Consultar Necesidad	77
Tabla 26: Casos de Uso - Eliminar Necesidad	78
Tabla 27: Caso de Uso - Crear Requerimiento.....	78
Tabla 28: Caso de Uso - Editar Requerimiento.....	79
Tabla 29: Caso de Uso - Consultar Requerimiento	79
Tabla 30: Caso de Uso - Eliminar Requerimiento.....	80
Tabla 31: Caso de Uso - Crear Caso de Uso	80
Tabla 32: Caso de Uso - Editar Caso de Uso	81
Tabla 33: Caso de Uso - Consultar Caso de Uso.....	81
Tabla 34: Caso de Uso - Eliminar Caso de Uso	82
Tabla 35: Casos de Uso - Generar Informe y Estadística.....	82
Tabla 36: Estructura de Pruebas de Usabilidad	84

RESUMEN

La calidad del software está tomando mayor importancia en las organizaciones por su influencia en los costes finales y como elemento diferenciador de la competencia y de la imagen frente a sus clientes. El desarrollo de este proyecto tuvo como principal objetivo diseñar un software para la gestión de requisitos en los niveles 1 y 2 de madurez de la norma ISO/IEC 15504 también conocido como Software Process Improvement Capability Determination, abreviado SPICE, en español, (Determinación de la Capacidad de Mejora del Proceso de Software) la cual se utilizó para mejorar y evaluar los procesos de desarrollo y mantenimiento de sistemas de información y productos de software en la empresa SOLINCES S.A.S.

Para tal fin la metodología que se empleó consistió en el desarrollo estándar Rational Unified Process (por sus siglas en inglés RUP) la cual provee un enfoque estructurado para realizar tareas y responsabilidades en una organización de desarrollo. Y su vez asegurar la producción de software de alta calidad, que cumpla las necesidades de sus usuarios finales, que sea realizado en las fechas acordadas y con el presupuesto disponible.

Los resultados de este proyecto fueron enfocados en apartes específicos, como el desarrollo de un software, teniéndose en cuenta la gestión de procesos que hace la empresa, realizando los correspondientes estudios en la aplicación de la norma ISO/IEC 15504 para evaluar y mejorar la capacidad y madurez de los procesos aumentando la satisfacción del cliente y a su vez se obtiene una importante ventaja respecto a la competencia. Al usar el modelo SPICE para la gestión de los procesos, se destaca el uso de una norma ISO internacional y abierta, el uso de ésta facilita la integración con otras del sector de las tecnologías de la información y comunicación (sector TIC), tales como: ISO 27000 de seguridad, ISO 20000 de servicios de IT, ISO 9000 e ISO 12207 la cual es específica para la ingeniería de software.

La finalidad de este proyecto fue mejorar los procesos en la gestión de requisitos en la empresa SOLINCES S.A.S lo cual posibilito enormemente la calidad de los procesos de desarrollo y mantenimiento de sistemas de información y productos de software en la empresa, redujo tiempo y costos, lo que genera dividendos de esta, mejorando el flujo de

trabajo, ya que todos los esfuerzos que se realizan en la empresa se verán evidenciados en un mejor producto.

Palabras clave: Calidad de software, gestión de proyectos, metodologías de desarrollo, modelo de evaluación, procesos software.

ABSTRACT

Software quality is taking more importance in organizations through its influence on the final costs as competitive differentiator and image to their clients. The development of this Project had as main objective to designing a software for the management of requirements in levels 1 and 2 of maturity of the ISO / IEC 15504 standard also known as Software Process Improvement Capability Determination, SPICE abbreviated in Spanish, (Determination of Capacity software process Improvement) which was used to improve and evaluate the processes of development and maintenance of information systems and software products in the company SOLINCES SAS.

To this purpose, the methodology used consisted of the standard development RUP (for its acronym in English, Rational Unified Process) which provides a structured approach to perform tasks and responsibilities in a development organization approach. And in turn ensure the production of high quality software that meets the needs of its final users, it is performed on the dates agreed with the available budget.

The results of this project were focused on specific asides, such as the development of a software, taking into account process management makes the company, performing related studies in the application of ISO / IEC 15504 standard for assessing and improving capacity process maturity and increasing customer satisfaction and in turn get a significant advantage over the competition. By using the SPICE model for the management of processes, the use of an open international standard ISO stands, the use of this facilitates integration with other technologies of information and communication technologies (ICT sector), such as : safety ISO 27000, ISO 20000 IT services, ISO 9000 and ISO 12207 which is specific to software engineering.

The purpose of this project was to improve the processes in requirements management in the enterprise SAS SOLINCES which greatly made possible the quality of the processes of development and maintenance of information systems and software products in the company, reduced time and costs, this generates dividends, improving workflow, and all efforts made in the company will be evidenced in a better product.

Keywords: Quality software, project management, development methodologies, evaluation model, software processes.

INTRODUCCIÓN

Día a día se habla del concepto calidad en diferentes entornos, ajeno a esto no es la ingeniería de software, donde constantemente se aumenta el interés por desarrollar productos de calidad; esto debido al alto nivel de exigencia por parte de los clientes y al rechazo que éstos tienen sobre productos que no cumplen con los requisitos propuestos o que no presentan una adecuada calidad (Alarcón Aldana, González Sanabria, & Rodríguez Torres, Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504, 2011, pág. 3).

Con base en lo anterior, internacionalmente se han propuesto distintos modelos, normas y estándares de calidad, con el fin de “suministrar los medios para que todos los procesos se realicen de la misma forma y sean una guía para lograr la productividad y la calidad” (Piattini & García, 2003); algunos de estos son Capability Maturity Model Integration (por sus siglas en inglés CMMI), la Organización Internacional de Normalización (ISO) y la Comisión Electrotécnica Internacional (IEC). ISO/IEC 12207 y ISO/IEC 15504.

En Colombia, adicional al desconocimiento de los modelos, normas y estándares de calidad, se evidencia una falta de recursos económicos y/o tecnológicos ocasionando que se presenten problemas de calidad en el desarrollo del software, y por ende generando un impacto negativo en el mercado internacional; sin embargo “con la desventaja competitiva que tiene la industria del software de Colombia, ésta aumenta progresivamente” (Pino, García, Ruiz, & Piattini, 2006).

El origen de esta investigación se basó en la falencia hallada en estudios anteriores en cuanto a la eficiencia del software a nivel de Colombia (Valencia, Villa, & Ocampo, 2009) (Ramírez Aguirre & Ramírez Arias, 2010), en la empresa SOLINCES S.A.S. se ha considerado que no se está llevando de la mejor manera el proceso de gestión de requisitos por lo tanto la funcionalidad no está acorde a las necesidades de los clientes.

Al percibir la necesidad que presentó la empresa SOLINCES S.A.S con las actividades de gestión de requisito, relación con los clientes, cumplimiento de calendarios, entre otras; se propuso el desarrollo de un software que facilite la interacción con los partes interesadas

dentro de un proyecto, en busca de la mejora de la calidad de los procesos de ingeniería de software y por lo tanto de los productos software.

Mediante el uso de la solución se hace un aporte en las mejoras y la madurez de los proyectos software los cuales cada vez más aportan a la automatización de operaciones en el medio social.

1. PLANTEAMIENTO DEL PROBLEMA

1.1. DESCRIPCIÓN DEL PROBLEMA

La industria del software ha adquirido importancia para mercados mundiales, por esta razón se hace necesario para entes productores llevar a cabo procesos de ingeniería de software que garanticen la obtención de componentes y productos finales de alta calidad, con el fin de competir por ganar la adjudicación de proyectos y un segmento de campo importante (ALARCON, 2011); para lograr dicho nivel de calidad las empresas se guían por cualquiera de las distintas metodologías o modelos de desarrollo existentes. Entre los modelos se encuentran SPICE (Software Process Improvement Capability Determination) y CMMI (Capability Maturity Model Integration), los cuales definen un catálogo, una colección estructurada, buenas prácticas, que describen las características de una actividad efectiva, como por ejemplo, pueden ser la gestión de configuración, la gestión de requisitos o el diseño de la arquitectura. Un modelo de proceso muy reconocido y utilizado es la norma ISO 12207.

Un modelo de evaluación proporciona los principios requeridos para aplicar calidad a la implantación de los procesos en una organización, esto se hace basado en modelos de calidad del proceso, los principales por su volumen de uso son: el CMMI propuesto por el Instituto de Ingeniería del Software (SEI), el cual es recomendado su uso para grandes empresas, y la norma ISO/IEC 15504 o modelo SPICE aplicado para pequeñas y medianas empresas (Inteco, 2008) (Valencia, Villa, & Ocampo, 2009), estos acompañan a la metodología de desarrollo que se requiera usar (Garzías, 2010). Dependiendo del alcance del proyecto se puede decidir entre metodologías ágiles, tales como: XP, SCRUM, ASD o AUP¹; y entre otras guiadas por la arquitectura, como son: los modelos en cascada, en espiral y RUP; llevando los procesos de la mano a la elaboración del software de alta competitividad con un costo conveniente (AENOR, 2009).

¹ Metodologías de Desarrollo de Software

La empresa SOLINCES S.A.S es una entidad encargada de desarrollar software para el manejo de control y gestión para la educación. Se encuentra ubicada en la ciudad de Cartagena en el Centro Médico los Ejecutivos, lleva alrededor de 12 años laborando y actualmente cuenta con una plataforma educativa donde interactúan más de 110 clientes del sector público, el grupo de trabajadores está distribuido con diferentes roles como son: un líder técnico, un líder de programación, dos analistas de requerimientos, dos programadores y por último 5 ingenieros para el proceso de pruebas².

Anteriormente SOLINCES S.A.S. contaba con una plataforma web alojada en la nube donde almacenan las aplicaciones, código y requerimientos de los proyectos software solicitados por los clientes, acompañados de las incidencias que estas presentan, sin embargo se consume mucho tiempo y recursos importantes para realizar las correcciones de estas incidencias, debido a problemas relacionados con los procesos de manejo de requisitos, específicamente con la gestión del modelo de ciclo de vida, puesto que falta establecer políticas y procedimientos para el despliegue de estos. Lo anterior genera entregas de productos posteriores a la fecha ante los clientes, problemas con la continuidad de los trabajos asignados a los desarrolladores y falta de organización en los modelos y aplicación de estándares.

La gestión de requisitos en SOLINCES S.A.S, se llevaba a cabo de manera regular, sin embargo, este proceso carece de la implementación de mejoras a nivel de organización, lo cual causa inconvenientes al momento de establecer y mantener la integridad de los elementos que conforman el producto software, dado que no se ponen a disposición permanente de las partes interesadas.

La empresa presentaba falencias en la comunicación entre las diferentes dependencias, generando inconsistencia entre lo que requiere el cliente, lo que pretende realizar el jefe de los proyectos y lo que producen los desarrolladores, también pasan por inconvenientes de valoración al momento de presupuestar los costos, estimar el tiempo necesario para las entregas y para asignar la cantidad de personal a cada tarea.

² Información suministrada por Giraldo Turizo Cano representante legal de SOLINCES S.A.S.

Para superar la problemática descrita anteriormente, la empresa usaba unas plantillas como guía en las cuales se almacenan los casos de uso, estos son el producto de la recolección de información en el proceso de requisitos, las plantillas son útiles para el inicio de la codificación, este método no cuenta con recomendaciones de calidad avalada por estándares internacionales a nivel de organización de procesos y mejoras, por lo tanto, no se asegura que los producto de trabajo y los proceso cumplan con las disposiciones y planes predefinidos.

Para llegar a una solución se realizó un sistema de gestión para el área de requisitos, el cual enfatizó en la relación con todas las partes interesadas en el desarrollo de software. Dado que SOLINCES S.A.S al igual que otras empresas posee áreas con necesidades que pueden ser suplidas mediante el uso del Modelo SPICE, se tomó como referencia este marco de trabajo, manejando recomendaciones útiles para la asignación correcta de calendarios, la gestión de requisitos, análisis de riesgo, orientación en los procesos de generación de modelos que tengan coherencia con los resultados obtenidos (Garzás & Piattini, 2010).

Al utilizar este sistema se llegó al desarrollo de productos de calidad fundamentados en la norma ISO/IEC 15504, lo cual tiene como beneficio, por la colaboración e interacción con el cliente un mayor acercamiento a los resultados deseados en cuanto a las tareas de identificación de requisitos como área funcional; ayudando a la identificación de riesgos, cumplimiento de calendarios, control de costos y mejoras en la calidad. Al usar el modelo SPICE para la gestión de los procesos, se destaca el uso de una norma ISO internacional y abierta, el uso de ésta facilita la integración con otras del sector de las tecnologías de la información y comunicación (sector TIC), tales como: ISO 27000 de seguridad, ISO 20000 de servicios de IT, ISO 9000 e ISO 12207 la cual es específica para la ingeniería de software (Garzás, 2010).

El proyecto de investigación basado en la norma ISO/IEC 15504 fortalece la comunidad científica teniendo acceso a la generación de nuevos conocimientos relacionados con la ingeniería del software, la gestión de la calidad de los productos y los procesos apoyados en la documentación completa de la investigación y del modelo SPICE. Mediante el uso de la solución se hace un aporte en las mejoras y la madurez de los proyectos software los cuales cada vez más aportan a la automatización de operaciones en el medio social.

1.2. FORMULACIÓN DEL PROBLEMA

¿Cómo aplicar los niveles de madurez 1 y 2 del modelo SPICE a empresas desarrolladoras de software usando una herramienta que facilite la interrelación con los clientes mejorando la gestión de requisitos?

2. JUSTIFICACIÓN DEL PROBLEMA

La industria del software en Colombia está en constante incremento, esto genera un mayor nivel de exigencia en cuanto a calidad, puesto que es un factor determinante para la competitividad de los proyectos software y la consecución de una buena posición dentro del mercado, obtener un buen nivel de calidad es importante para las empresas, las cuales deben establecer normalización sobre el procesos de desarrollo, según Carlos Manuel Fernández gerente de AENOR que es una entidad certificadora en España, dichos procesos son usados por una organización para planificar, gestionar, ejecutar, monitorizar, controlar y mejorar sus actividades en el desarrollo del software (Fernández, 2012).

Existen diversos modelos que se usan en la calidad de los procesos, los dos más conocidos a nivel mundial son CMMI y el propuesto por la ISO la norma ISO/IEC 15504 (Inteco, 2008), el primero es el más difundido pero su aplicación se centra en grandes empresas, el segundo está adaptado para pequeñas y medianas empresas.

Al percibir la necesidad que tiene la empresa SOLINCES S.A.S con las actividades de gestión de requisito, recolección de la información, relación con los clientes, cumplimiento de calendarios y control de costos; se propuso el desarrollo de un sistema que facilite la interacción con los stakeholders (partes interesadas dentro de un proyecto), en busca de la mejora de la calidad de los procesos de ingeniería de software y por lo tanto de los productos software.

La implementación de este proyecto ayudó en la búsqueda de la madurez y mejoras de los procesos en los niveles 1 y 2 propuestos por el modelo SPICE, adicional a esto cabe mencionar que al usar una norma internacional y abierta, propuesta por la ISO, siendo la organización de mayor importancia a nivel mundial en estandarización, tendrá como beneficios adicionales la compatibilidad con otras normas de gran importancia en las tecnologías de información y comunicación: en seguridad la ISO 17799, para servicios de IT la ISO 20000 y la ISO 9000, entre otras.

La viabilidad económica de este proyecto se puede evidenciar, considerando que la empresa SOLINCES S.A.S, no ha afirmado sus procesos en estándares internacionales, debido a que esto requiere de una gran inversión en tiempo y dinero, además de la alta complejidad de las recomendaciones y la remuneración de dicha inversión sería a largo plazo (Piattini, Pino, & Gracia, 2006). El desarrollo de este sistema no requiere de una gran inversión y estará centrado en facilitar la gestión de requisitos como área funcional, mediante el modelo SPICE y adaptándose a cualquier metodología de desarrollo.

Para llevar a cabo este proyecto se hizo un estudio minucioso de cada componente de la norma ISO/IEC 15504, y se obtuvo un producto que conste de un marco de procesos comunes que se puedan aplicar en la realización de proyectos software en el área de gestión de requisitos (Lee, Kim, & Hwang, 2001) (ACIS, 2007), para esta solución se aplicó los conocimientos adquiridos durante el transcurso de la carrera Ingeniería de Sistemas en asignaturas de la línea de desarrollo de software que van desde: algoritmos y programación básica hasta tópicos avanzados de ingeniería del software, haciendo especial énfasis en las asignaturas de ingeniería de servicios de internet, ingeniería del software, gestión de la calidad prueba y mantenimiento de software y gestión de proyectos de ingeniería del software, el proyecto se fundamenta en el grupo de investigación E-Soluciones grupo de investigación en soluciones tecnológicas de nueva generación para el desarrollo social, industrial y empresarial en la línea de investigación E-Servicios.

El software tendrá un impacto en la sociedad, puesto que inicialmente se desarrolló para la empresa SOLINCES S.A.S, y este proyecto implica un mejoramiento significativo orientado a mejorar la calidad de los procesos de la industria del software.

3. MARCO DE REFERENCIA

3.1. ESTADO DEL ARTE

Desde sus inicios el software ha ido ganando terreno en todos los campos de la vida cotidiana, cada vez hay más exigencias en cuanto al desarrollo y la gestión de proyectos de tipo informático, esto conlleva a que los entes desarrolladores busquen estandarizar las actividades y los procesos con el fin de obtener productos de calidad. Existen organismos internacionales que han dedicado sus esfuerzos en la creación de estándares para la industria del software, entre estos se encuentra el Instituto de Ingeniería de Software (SEI) y la Organización Internacional para Estándares (ISO).

A nivel internacional cada una de estas organizaciones ha dejado un gran aporte en cuanto a estandarización de procesos, por un lado el SEI implementó el modelo CMMI (CMMI Institute, 2010), mientras que la ISO desarrolló una norma conocida como la ISO/IEC 15504.

La investigación se limitó a los aspectos relacionados con la gestión de requisitos, puesto que cubrir otros procesos de la ingeniería de software se hace demasiado extenso y abarcar niveles superiores al básico y el gestionado. Es apropiado saber que el nivel 1 de madurez consta de tres procesos, que son: suministro, definición de los requisitos de usuario y análisis de los requisitos del sistema, así el nivel 2 abarca los procesos de la gestión del modelo de ciclo de vida (Rodriguez, Alonso, & Sanchez, 2005). Para los niveles superiores se requiere de periodos más extensos y estuvieron por fuera del alcance y tiempo del proyecto (ALARCON, 2011).

3.1.1. HERRAMIENTAS EXISTENTES PARA LA GESTIÓN DE PROYECTOS SOFTWARE

En cualquier proyecto de desarrollo de software, es necesario disponer de una gestión eficaz para garantizar que el sistema cumpla los objetivos y que se desarrolle dentro de un presupuesto aceptable.

Existen herramientas que desempeñan un papel fundamental en obtener sistemas que funcionen. En la Tabla 1, 2 y 3 se ilustran diferentes tipos de herramientas para la gestión de proyecto software como son de tipo Web, de escritorio y código abierto, con varias actividades o beneficios las cuales se tuvieron en cuenta para el desarrollo de esta investigación (Gonzalo, 2012).

Tabla 1: Herramientas WEB

Herramientas WEB				
Beneficios	Achievo	TeamWork	eGroupWare	Trac
Seguimiento de tareas	x	x	x	x
Control de calendarios	x	x	x	x
Interacción cliente-proveedor		x	x	
Generación de informes	x		x	x
Asignación de personal		x		
Integra diseño y gestión				
Incluye manual de usuario		x	x	x
Gratis	x	x	x	x
Pago	x		x	
Multi Idiomas	x	x	x	x

Tabla 2: Herramientas de Escritorio

Herramientas de Escritorio				
Beneficios	GanttProject	Gantt pv	Planner	Microsoft Project
Seguimiento de tareas	x	x	x	x
Control de calendarios	x	x	x	x
Interacción cliente-proveedor				
Generación de informes	x	x	x	x
Asignación de personal	x			
Integra diseño y gestión				
Incluye manual de usuario	x	x		x
Gratis	x	x	x	x
Pago				
Multi Idiomas			x	x
Multiplataforma	x	x	x	x

Tabla 3: Herramientas de Código Abierto

Herramientas de Código Abierto				
Beneficios	Colabtive	Tutos	RedMine	Todayu
Seguimiento de tareas	x	x	x	x
Control de calendarios	x	x	x	x
Interacción cliente-proveedor				x
Generación de informes	x	x	x	
Asignación de personal		x		x
Integra diseño y gestión			x	
Incluye manual de usuario	x	x		x
Multi Idiomas		x	x	

3.1.2. LISTADO DE EMPRESAS QUE APLICAN EL MODELO SPICE

Esta es una recopilación de empresas que hasta la fecha han obtenido un certificado de calidad y son reconocidas por haber aplicado dentro de sus procesos la norma ISO/IEC 15504 SPICE. La mayoría de las empresas son españolas, siendo este el primer país europeo en calidad del proceso de software con un crecimiento de más del 70% en 5 años (Garzías, 2011), a continuación se ilustra la tabla 4, con relación a las empresas con aplicación al modelo SPICE.

Tabla 4: Empresas certificadas por el modelo SPICE

Empresa	País	Nivel	Año de certificación
Alhambra Eidos	España	2	2010
Altana	España	2	2010
Amplia Soluciones	España	2	2010
Bitware	España	2	2012
CAF Power and Automation	España	2	2012
Dimetronic	España	2	2011
GCL Informática	España	2	2010
HSM Soft	España	2	2012
Indenova	España	3	2010
Innova Desarrollos Informáticos	España	2	2010
Isi Diseños de Sistemas de Información	España	2	2010
Mangana webs	España	2	2010
Net2u	España	3	2009
NYCE	México	2	2010
TB-Solutions	España	3	2009

Fuente: (AENOR, 2012)

3.1.3. LISTADO DE EMPRESAS QUE APLICAN EL MODELO CMMI

La industria del software se distingue por estar conformada por empresas importantes que se preocupan por la calidad de sus productos y servicios. (CSOFTMTY Concejo de software de Nuevo Leon, 2009).

A continuación en la tabla 5, presentamos un listado de algunas empresas certificadas por el modelo CMMI en Latinoamérica.

Tabla 5: Empresas certificadas por CMMI en Latinoamérica

Empresa	Sitio Web	Dirección	Nivel CMMI	País
AD Infinitum	adinfi.com	Roma #635, Col. Mirador Monterrey, N.L.	Nivel 3	México
Cibernet	cibernet.com.mx	Puerto México #240, Col. Valle de las Brisas Monterrey, N.L.	Nivel 2	México

Para el caso de la aplicación del modelo SPICE en Colombia se encuentran estudios acerca de la adaptación de las normas ISO/IEC 12207 e ISO/IEC 15504 para la madurez de procesos software en países en desarrollo (Piattini, Pino, & Gracia, 2006), existen empresas que implementan una metodología de mejora de procesos que les permita concluir sus proyectos a tiempo con calidad y rentabilidad.

A continuación en la tabla 6, se ilustra lista de algunas empresas colombianas certificadas en los niveles 3, 4 y 5 por modelo CMMI.

Tabla 6: Empresas Certificadas por CMMI en Colombia

Empresa	Sitio Web	Dirección	Nivel CMMI	Área	Fecha de Certificación
Red Colombia S.A.	http://www.redcolombia.com.co/	Bogotá, Colombia Avda. 13 # 118 - 68	Nivel 3	Unidad de Integración de Soluciones de Red Colombia S.A	Septiembre 15, 2006
Asesoftware Limitada	http://www.asesoftware.com/web/es/	Bogotá, Colombia Cr13 93-68 Of 407	Nivel 3	Software Development Division at Asesoftware Limit	Diciembre 16, 2006
Open Systems S.A.	http://www.opensystems.cl/presentacion.html	Córdoba 2525 Piso PA, Rosario - SANTA FE	Nivel 4	Centro de Desarrollo de Software: (Software Development Center)	Mayo 13, 2006
Productora de Software (PSL)	http://www.psl.com.co/	Itagüí, Colombia Cra 42 (Autopista sur), No. 72-11 Piso 4	Nivel 5	Productora de Software (PSL) – Medellín Software Engineering Unit	noviembre 30, 2007

Fuente: (Cañon, 2011)

A nivel local en Cartagena de Indias las consultas realizadas en el Registro Único Empresarial de la Cámara de Comercio y posteriormente en las empresas halladas se obtuvo como resultado que organizaciones pequeñas similares a SOLINCES S.A.S, como es el caso de Mimetics S.A.S ubicada en Cartagena Av. Pedro de Heredia 18 B 74 con 3 años de

experiencia, están trabajando con gestión de calidad a nivel personal como Personal Software Process (por sus siglas en ingles PSP) y a nivel de equipo Team Software Process (por sus siglas en ingles TSP), como base para aplicar un modelo a nivel de organización (SPICE o CMMI)³.

³ Información obtenida vía telefónica, suministrada por personal de la empresa Mimetics S.A.S.

3.2. MARCO TEÓRICO

Para hablar de un sistema para la gestión de software es importante conocer términos como: gestión de proyectos, calidad del software, metodologías de desarrollo y modelos de evaluación como SPICE y CMMI, niveles de organización como Team Software Process (por sus siglas en inglés TSP) Personal Software Process (por sus siglas en inglés PSP) para saber qué papel cumplen dentro del proyecto y cómo se pueden relacionar.

3.2.1. GESTIÓN DE PROYECTOS

La gestión de proyecto es el proceso por el cual se planifica y controla el desarrollo de sistemas aceptables y dentro de un periodo de tiempo planeado. Aunque las herramientas y técnicas del análisis y el diseño del sistema desempeñan un papel fundamental en obtener sistemas que funcionen, estos métodos no son suficientes por sí mismo. Una mala gestión de proyecto puede dar al traste por los mejores métodos de análisis y diseño de proyectos o hacerlos ineficaces (Kendall & Kendall, 2002). La gestión de un proyecto de software se centra en cuatro partes como son: el personal, el producto, el proceso y el proyecto (Pressman, 2010).

El personal

El factor humano es importante en la ingeniería de software. Es importante tener la capacidad de gestión del personal con el fin de aumentar la preparación en la organización del software; ayudando a atraer, motivar y retener el talento necesario para mejorar su capacidad de desarrollo de software. Las organizaciones que alcancen la madurez en el área de gestión de personal tienen una mayor probabilidad de implementar prácticas más eficaces de ingeniería de software, esto guía a que las organizaciones tengan un proceso de software maduro. (Darwin Jimenez Garzon, 2007)

El producto

Para pensar en un proyecto es importante antes definir cuál es el ámbito del producto que se desea realizar, junto a este ámbito se deben tener en cuenta los objetivos del producto, se

deben definir e identificar restricciones, mediante el uso de esta información se pueden estimar los costos, la valorización efectiva de los riesgos y la modulación real de las tarea a realizar. El equipo de desarrollo y el cliente se reúnen para determinar dicho amito con el fin de garantizar la viabilidad del proyecto (Pressman, 2010).

El proceso

El proceso de software es el que suministra el marco de trabajo, y ante todo el proceso define el conjunto de tareas que se realizarán, la descomposición de estas, los tiempos establecidos, los hitos, los productos de trabajo, los puntos de control y la adaptación del marco de trabajo a las características del proyecto software, es aquí donde juega un papel importante el modelo de procesos el cual está incluido en todas las actividades y durante todo el proceso (Pressman, 2010).

El proyecto

El proyecto enmarca todo lo mencionado antes, para entenderlo se debe entender cada una de las partes: el ámbito del producto, el personal involucrado, el tiempo estimado, las tareas a realizar, además de esto se debe saber qué puede salir mal y cómo poder adelantar actividades que no afecten a otras. El proyecto se realiza de una manera controlada y planificada encaminado a un objetivo principal, para que esto sea exitoso los gestores deben tener un enfoque común, para planificar, supervisar y controlar las actividades guiándose a los casos de éxito (Pressman, 2010).

PMBOK

Para llevar a cabo de manera eficaz la gestión de proyectos se han planteado normas para la dirección de estos, la más difundida y una de las más completas es PMBOK (Project Management Body of Knowledge) en español Guía de los Fundamentos para la Dirección de Proyectos, es un estándar reconocido en la profesión de la gestión de proyectos (no sólo de desarrollo de software) desarrollada por el PMI (Project Management Institut).

El PMBOK es una recopilación de procesos y áreas del conocimiento que son aceptadas como buenas prácticas dentro de la gestión de proyectos, es un estándar reconocido

internacionalmente (IEEE Std 1490-2003) que provee los fundamentos de la gestión de proyectos que son aplicables a un amplio rango de proyectos, incluyendo construcción, software, ingeniería, etc. (PMI Project Management Institute, 2004).

3.2.2. CALIDAD DEL SOFTWARE

La calidad del software se mide a través de algunos factores que se pueden dividir en externos e internos. Los externos son los que ve el usuario final del software, como son la eficiencia, la corrección, la facilidad de uso, etc., indispensables en cualquier solución.

Los factores internos de calidad son aquellos que vienen dados por la estructura misma del código, y sólo los ve el ingeniero de software. Entre otros, se pueden mencionar la documentación, la claridad del código, la modularidad, etc. Pero, considerando los grandes problemas y costos asociados con el mantenimiento, se puede afirmar que el principal factor de calidad interna de un programa es la facilidad que éste tiene para evolucionar, lo cual está fuertemente relacionado con su arquitectura interna. A esta característica del software se le denomina extensibilidad (Villalobos, 1993).

Factores de calidad

Los factores desarrollados según el modelo de McCall, se centra en tres aspectos importantes de un producto de software: sus características operativas (corrección, fiabilidad, eficiencia, integridad, facilidad de uso), su capacidad para soportar los cambios (facilidad de mantenimiento, facilidad de prueba, flexibilidad) y su adaptabilidad a nuevos entornos (portabilidad, reusabilidad, facilidad de interoperación) (McCall, Richards, & Walters, 1977).

3.2.3. METODOLOGÍAS DE DESARROLLO

Una metodología es un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea y abierta cada una de las actividades del ciclo de vida de un proyecto de desarrollo. Es un proceso de software detallado y completo (INTECO, 2009).

Una metodología de desarrollo de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información. Una gran variedad

de estos marcos de trabajo han evolucionado durante los años, cada uno con sus propias fortalezas y debilidades. Una metodología de desarrollo no tiene que ser necesariamente adecuada para usarla en todos los proyectos, cada una de las técnicas disponibles es más adecuada dependiendo del alcance y el tamaño de estos, basados en consideraciones técnicas y organizacionales del equipo de trabajo.

3.2.4. MODELOS DE PROCESOS DE SOFTWARE

Un modelo de procesos define un catálogo, una colección estructurada, buenas prácticas que describen las características de un proceso efectivo (mientras que el modelo de evaluación, es decir, la ISO 15504, proporciona los principios requeridos para realizar una evaluación de la implantación de dicho modelo de procesos en una organización). Un modelo me indica el tipo de proceso que se puede encontrar en una organización que desarrolla software con calidad (Garzás, 2010).

Para que una organización mejore la calidad de sus productos debe tener un método probado, consistente y fiable para evaluar el estado de sus procesos y además, unos medios para usar los resultados de la evaluación como parte de un programa de mejora coherente. El proyecto internacional SPICE, llevado a cabo por la organización ISO, ha obtenido en su primera fase del proyecto un Informe Técnico Tipo 2 (ISO 15504) formado por un conjunto de documentos todos ellos bajo el título general de Evaluación del Proceso Software. (De Amescua, Lloréns, & García, 2008)

Por otra parte actualmente se han desarrollado modelos para la gestión de los procesos de software a nivel mundial con el fin de mejorar la productividad y la calidad. Uno de los modelos más difundidos es Modelo de Capacidad y Madurez Integrado (Capability Maturity Model Integration - CMMI).

Este proporciona una orientación sobre el modo de realizar gestión a los procesos se clasifican en cuatro categorías ingeniería, gestión de proyectos, gestión de procesos y soporte, a la vez que se logra una evolución hacia la cultura de la ingeniería y una mejora continua. (Universidad Tecnológica de Pereira, 2009).

3.2.5. MODELOS DE EVALUACIÓN DE PROCESOS

Un modelo de evaluación de procesos es una norma para establecer y mejorar la capacidad y madurez de los procesos de las organizaciones, estos son usados para evaluar de manera general cualquier modelo de procesos, es útil en cualquier tipo de proyectos y se aplican directamente a las organizaciones.

Entre los modelos para la evaluación de proceso encontramos los siguientes:

Modelo SPICE

La ISO/IEC 15504, conocida como SPICE (Software Process Improvement and Capability dEtermination) es un modelo de evaluación y mejora de los procesos de desarrollo y mantenimiento de sistemas y productos de software. El estándar ISO 15504 es una herramienta que ayuda a reducir costes y mejorar la calidad evitando problemas. (Asociación Española para la Calidad, 2013).

ISO/IEC 15504 proporciona un marco para la evaluación de capacidad de proceso. Este marco puede ser usado por organizaciones complicadas en la planificación, la dirección, la supervisión, el control y el mejoramiento la adquisición, el suministro, el desarrollo, la operación, la evolución y el apoyo de productos y servicios.

La ilustración 1, sintetiza la dimensión de la capacidad de proceso, a continuación, se describe cada nivel.



Ilustración 1: Niveles de Madurez de SPICE

La dimensión de la capacidad del proceso es definida por una escala jerárquica de tres niveles, que representan el incremento de las capacidades de los procesos de desarrollo de software. La escala queda definida de menor a mayor grado de capacidad por los siguientes niveles:

- Nivel 0. Proceso Incompleto. El proceso no está implementado o fallan los logros de sus propósitos. Hay poca o ninguna evidencia de algún logro sistemático del propósito del proceso. Hay grandes fallas que limitan o incluso impiden el cumplimiento de los objetivos y propósitos del proceso. Hay muy poco o incluso ningún, producto y/o salida identificados a lo largo del proceso.
- Nivel 1. Proceso Realizado. La implementación del proceso logra su propósito de proceso. El propósito del proceso es generalmente alcanzado, aunque este no sea siempre planificado o controlado. Los individuos dentro de la organización reconocen que se debe llevar a cabo una acción la cual ejecuta cuando es requerida. Existen productos generados por el proceso, utilizados para medir el logro de objetivos.
- Nivel 2. Proceso Gestionado. A la realización del proceso se le implementa una manera de gestionarlo (se planea, se monitorea y se ajusta). Sus productos de trabajo

se establecen, controlan y mantienen apropiadamente. El proceso genera productos capaces de ser liberados en tiempo y bajo planes controlables. Los productos generados están alineados con determinados procesos que se encuentran en este nivel cumplen con ciertas especificaciones puntuales de calidad respetando un cronograma y un plan. (Pino, García, Ruiz, & Piattini, 2006).

Procesos de los niveles 1 y 2 de madurez

- Suministro (SUM)
- Definición de Requisitos de los Stakeholders (RQU)
- Análisis de los Requisitos del Sistema (RQSYS)
- Planificación del Proyecto (PP)
- Evaluación y Control del Proyecto (ECP)
- Gestión del Modelo de Ciclo de Vida (MCV)
- Gestión de la Configuración (GC)
- Gestión de la Configuración del Software (GCS)
- Medición (MED)
- Aseguramiento de la Calidad del Software (ACS)

Entre los principales beneficios de una implantación de SPICE, podemos destacar los siguientes:

- Reconocimiento en el mercado, siendo un factor diferenciador ante su competencia.
- Aumenta la satisfacción de sus clientes.
- Mejora interna de su empresa:
 - Implantación de una metodología común a todas las áreas.
 - Control de todas las fases y áreas de gestión de un proyecto software.
 - Detectar y corregir posibles fallos en cada etapa de un proyecto.
 - Aumento de los beneficios del proyecto.

Visión General de SPICE

El marco para la evaluación de procesos software se puede utilizar por organizaciones implicadas en la planificación, gestión, monitorización, control y mejora de la adquisición, suministro, desarrollo, operación, evolución y soporte del software.

La evaluación del proceso examina los procesos utilizados por una organización para determinar si son efectivos para conseguir sus objetivos. Los resultados de la evaluación se pueden utilizar para conducir las actividades de mejora o para la determinación de la capacidad del proceso. (de Amescua, Lloréns, & García)

Elementos principales de SPICE

Los elementos esenciales para comprender SPICE son los siguientes:

Modelo de Referencia

Los resultados de la evaluación del proceso se describe en un modelo de dos dimensiones: Dimensión del proceso y Dimensión de la capacidad. Esto es lo que se denomina arquitectura del modelo de referencia.

1. **Dimensión del proceso:** que está caracterizado por los objetivos del proceso que constituye los elementos fundamentales a medir;
2. **Dimensión de la capacidad del proceso:** que está caracterizado por una serie de atributos de proceso, aplicables a cualquier proceso, que representan características necesarias para gestionar y mejorar su capacidad de realización.

Dimensión del proceso: El modelo de referencia agrupa los procesos en categoría de procesos. Estos procesos se corresponden con los definidos en *ISO 12207 Software Life-Cycle Process*, (tabla 1). La descripción de cada proceso consta de:

- una declaración del objetivo del proceso describiendo a un alto nivel los objetivos generales del proceso, y también una descripción en términos

genéricos de los probables resultados de una implementación efectiva del proceso; y

- o una o más observaciones proporcionando mayor información sobre los procesos y su relación a los procesos definidos en ISO 12207 y a otros procesos en este modelo de referencia.

Dimensión de la capacidad: Está formada por seis niveles de capacidad y nueve atributos de proceso. Un nivel de capacidad está formado por uno o varios atributos que conjuntamente proporcionan una mejora importante en la capacidad de realizar un proceso. Los niveles proporcionan una manera racional de progresar en la mejora de la capacidad de cualquier proceso.

El modelo de referencia por sí solo no se puede usar como base para realizar evaluaciones fiables y consistentes de la capacidad de proceso puesto que no es suficiente el nivel de detalle. Se debe soportar con un conjunto de indicadores comprensibles del rendimiento y capacidad del proceso.

Tabla 7: Dimensión del proceso

Categoría de Proceso		Proceso	
ID	Nombre	ID	Nombre
CUS	Cliente-Suministrador		
<i>Consta de los procesos que directamente impactan al cliente, soportan el desarrollo y la entrega, al cliente, del software proporcionándole su correcta operación y utilización.</i>		CUS.1	Adquirir software
		CUS.2	Gestionar las necesidades del cliente
		CUS.3	Suministrar software
		CUS.4	Operar software
		CUS.5	Proporcionar servicio al cliente
ENG	Ingeniería		

<p><i>Consta de los procesos que directamente especifican, implementan, o mantienen un sistema, el producto software y su documentación de usuario.</i></p>		ENG.1	Desarrollar los requisitos y diseño del sistema
		ENG.2	Desarrollar los requisitos software
		ENG.3	Desarrollar el diseño software
		ENG.4	Implementar el diseño software
		ENG.5	Integrar y probar el software
		ENG.6	Integrar y probar el sistema
		ENG.7	Mantener el sistema y el software
SUP	Soporte		
<p><i>Consta de los procesos que pueden ser empleados por cualquiera de los otros procesos (incluyendo los procesos de soporte) en distintos puntos del ciclo de vida del software.</i></p>		SUP.1	Desarrollar documentación
		SUP.2	Realizar la gestión de la configuración
		SUP.3	Realizar el aseguramiento de la calidad
		SUP.4	Realizar la verificación del producto
		SUP.5	Realizar la validación del producto
		SUP.6	Realizar revisiones conjuntas
		SUP.7	Realizar auditorias
		SUP.8	Realizar resolución de problemas
MAN	Gestión		
<p><i>Consta de los procesos que contienen prácticas de una naturaleza genérica que pueden ser utilizadas por cualquiera que gestione cualquier clase de proyecto en un ciclo de vida software.</i></p>		MAN.1	Gestionar el proyecto
		MAN.2	Gestionar la calidad
		MAN.3	Gestionar los riesgos
		MAN.4	Gestionar los subcontratistas

ORG	Organización		
<i>Consta de los procesos que establecen los objetivos del negocio de la organización y del producto de proceso de desarrollo, y activos de recursos que, cuando son empleados por los proyectos en la organización, ayudarán a la misma a lograr sus objetivos de negocio.</i>	ORG.1	Ingeniería del negocio	
	ORG.2	Definir el proceso	
	ORG.3	Mejorar el proceso	
	ORG.4	Proporcionar recursos humanos con habilidades	
	ORG.5	Proporcionar la infraestructura de ingeniería del software	

Ventajas y desventajas

SPICE ofrece una base para una evaluación muy detallada del estado actual del proceso de una organización. Por su gran nivel de descomposición de los procesos e indicadores, proporciona evaluaciones objetivas y con resultados repetibles, especialmente cuando es realizada por evaluadores entrenados y cualificados. El Instituto Europeo de Software abreviado por sus siglas en inglés ESI (European Software Institute) ya ofrece cursos para ello.

Al disminuir la subjetividad se consigue reducir discordias sobre los resultados de la evaluación y a adoptar actitudes positivas de los equipos hacia la evaluación.

Por contra se requiere un gran esfuerzo para realizar las evaluaciones y por tanto un alto coste. Las evaluaciones se pueden llevar a cabo por personal interno de tal manera que se puedan ver reducidos estos costes. Es importante tener en cuenta que la evaluación no necesita abordarse a toda la organización, las evaluaciones SPICE se puede realizar únicamente en aquellos procesos que sean áreas de problema.

El modelo de referencia SPICE no contiene una estrategia de mejora del proceso. Esto puede verse como positivo o negativo dependiendo de lo que se quiera.

Pero la ventaja principal es que al disponer de un estándar internacional se pueden realizar comparaciones a nivel mundial entre evaluaciones en contextos similares.

Familia Normas ISO 15504

El rol de la norma es proveer un marco de referencia que determine las fortalezas y debilidades de los procesos, así como también mejorar los procesos de software y medir sus mejoras, además de ser un marco de referencia para aquellos que pretenden adquirir un sistema, puesto que permite evaluar la capacidad de los proveedores de sistemas y determinar los riesgos de negocio para una empresa que considera desarrollar un nuevo producto o servicio de software. (Software, 2012).

La norma está orientada a los procesos y se integra por las siguientes cinco partes.

- ISO/IEC 15504-1:2004 Conceptos y Vocabulario.

Representa una introducción general a la norma, proporcionando una guía de la misma. En esta parte se incluye el conjunto de términos definidos específicamente para la norma.

- ISO/IEC 15504-2:2003 Realizando una Evaluación (Requisitos, normativa).

Define los requisitos que deben cumplir una evaluación para que produzca resultados repetibles, fiables y consistentes.

- ISO/IEC 15504-3:2004 Guía para Realización de Evaluaciones.

Establece una guía para la realización de evaluaciones de procesos, interpretando los requisitos de las partes normativas para diferentes contextos de evaluación.

- ISO/IEC 15504-4:2004 Guía para el Uso de Resultados de Evaluaciones.

Proporciona una guía para poder utilizar los resultados de una evaluación en la mejora de los procesos evaluados. La guía incluye ejemplos de la aplicación de mejoras en una gran variedad de situaciones.

- ISO/IEC 15504-5:2008 Un Modelo de Evaluación de Procesos Ejemplar.

Proporciona un modelo totalmente compatible con la parte de la norma que incluye un conjunto de indicadores que facilitan el cálculo de la capacidad de los procesos.

En resumen, la implantación de SPICE en su empresa le permitirá abordar cualquier tipo de proyecto software siguiendo unas directrices comunes en todos ellos.

SPICE proporciona todas las facilidades para la evaluación del proceso y establece los requisitos mínimos para realizar una evaluación que asegure la repetitividad y consistencia de las valoraciones obtenidas.

Después de su ejecución, se debe obtener la información relevante de cada proceso, y el punto hasta el cual estos cumplen con su propósito.

Es un Marco de referencia para:

- Determinar las fortalezas y debilidades de los procesos.
- Mejorar los procesos de software y medir sus mejoras.
- Aquellos que adquieren un sistema para evaluar la capacidad de los proveedores de sistemas.
- Determinar los riesgos de negocio para una empresa que considera desarrollar un nuevo producto de software o servicio.

Gestión de Requisitos

La gestión de requisitos en la Ingeniería de Software, se ha constituido en una de las principales estrategias para garantizar desde las primeras etapas del proceso de desarrollo de software, la calidad de las aplicaciones resultantes además implica establecer un entendimiento compartido entre los stakeholders (personas relacionadas con el proyecto) y

los requisitos que ellos han especificado para ser incluidos en el producto de software (Puello Marrugo & Garcia Ramirez, 2009).

¿Por qué necesitamos requisitos?

El coste de una buena recogida de requisitos y análisis del sistema a desarrollar es menor comparado con el coste resultante de tener requisitos pobres, es decir, el coste de reparar productos deficientes o de poca calidad, el coste de los proyectos cancelados y el coste de haber perdido la oportunidad de tener el producto correcto en el momento correcto. El fundamento básico de cualquier software recae sobre su proceso de ingeniería de requisitos. El éxito o fallo del software depende casi siempre de cómo de bien se hayan capturado, entendido y usado los requisitos como base para el desarrollo. La ingeniería de requisitos es la fase de la ingeniería del software donde se definen las propiedades y la estructura del software. La ingeniería de requisitos comprende el desarrollo y gestión de requisitos. (INTECO, 2008)

- El desarrollo de requisitos implica entender los requisitos de negocio, identificar los requisitos de usuario y trasladar los requisitos de usuario y de negocio a requisitos de sistema/software.
- La gestión de requisitos implica gestionar los cambios de requisitos y mantener la consistencia entre los requisitos y otros productos de trabajo del proyecto.

¿Qué es un requisito?

Un requisito es algo que el producto debe hacer o una característica que debe tener. Un requisito existe por el tipo de demanda que tiene el producto o porque el cliente quiere que el requisito sea parte del producto entregado. La tarea de todo analista de requisitos es hablar con la gente, entenderla, escuchar lo que dicen y también lo que no dicen, para entender lo que necesitan.

La tabla 8 muestra los diferentes tipos de requisitos y descripción que se identifican (INTECO, 2008)

Tabla 8: Tipos de requisitos

Tipos de requisitos	Descripción
Requisitos de negocio	Dan una descripción a alto nivel de lo que el sistema debe hacer.
	Representan: los objetivos, la base del negocio, estrategias, visión, alcance y el valor esperado del desarrollo del software
Requisitos de usuario	Son una descripción de las tareas que el sistema ha de ejecutar cuando el usuario opera con él.
	Describen la funcionalidad necesaria para satisfacer tareas específicas, necesidades operacionales y grupos de usuarios.
Requisitos del sistema/software	Definen las funcionalidades y características que debe tener el sistema para satisfacer tanto los requisitos de negocio como los de usuario.
	Van a servir como base para llevar a cabo la arquitectura, diseño y planes de pruebas del sistema
Restricciones	Son condiciones que limitan las elecciones disponibles al diseñador o programador. Pueden ser restricciones del propio proyecto o del diseño del producto.

Desarrollo de Requisitos

Las principales actividades realizadas durante el proceso de desarrollo de requisitos son las que se muestran en la ilustración 3.

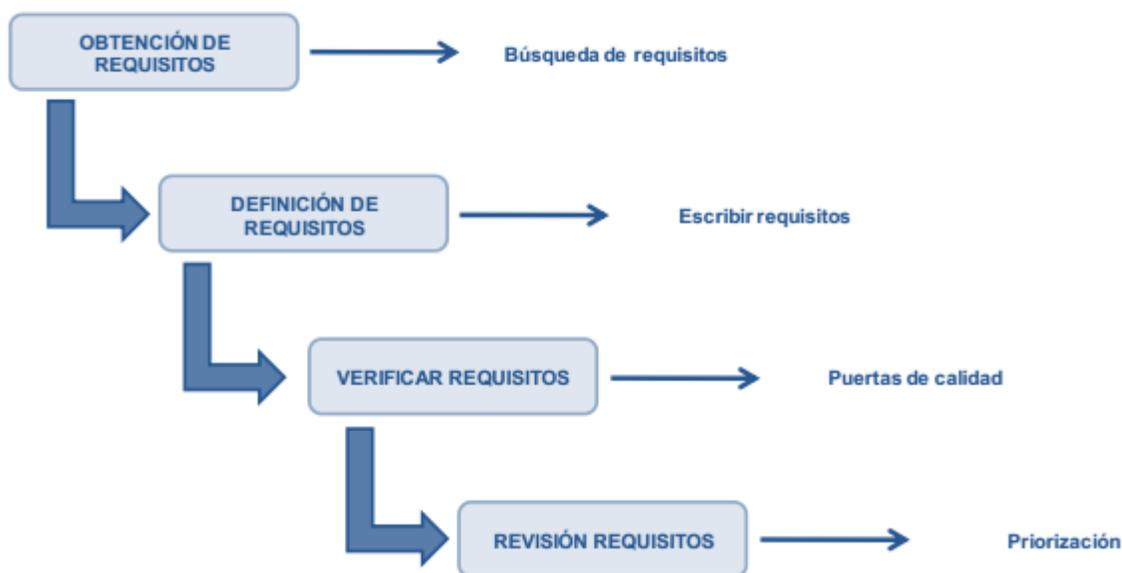


Ilustración 2: Desarrollo de requisitos

Enfoque de Algunos Modelos

Tanto la gestión como el desarrollo de requisitos son procesos contemplados por los principales modelos de mejora de procesos orientados al desarrollo de software.

CMMI y SPICE son modelos de mejora de procesos que describen los procesos que una organización debe ejecutar para la adquisición, desarrollo y mantenimiento de productos y servicios software. Ambos modelos contemplan, entre sus áreas de proceso, la gestión y desarrollo de requisitos. Para implementar correctamente esta área, ambos modelos proponen una serie de prácticas a seguir (INTECO, 2008).

Según CMMI, el propósito de la gestión de requisitos es gestionar los requisitos de los productos del proyecto y de sus componentes e identificar las inconsistencias entre requisitos,

planes del proyecto y productos de trabajo. Las prácticas que propone para llevarlo a cabo son:

- Obtener un entendimiento de requisitos
- Obtener un compromiso con los requisitos
- Gestionar los cambios de los requisitos
- Mantener una trazabilidad bidireccional de los requisitos
- Identificar inconsistencias entre los requisitos y los productos de trabajo del proyecto.

CMMI también contempla el desarrollo de requisitos. Su propósito en aquí es producir y analizar requisitos de usuario, de producto, y de los componentes de producto. Las metas que habría que cumplir para satisfacer esta área de proceso son:

- Desarrollar los requisitos de cliente
- Desarrollar los requisitos de producto
- Analizar y validar los requisitos

Por otra parte, los procesos del modelo SPICE que contemplan las actividades de desarrollo y gestión de requisitos son:

- ENG.1 Recogida de requisitos: recoger, procesar y registrar las necesidades y requisitos cambiantes del cliente a lo largo de la vida del producto y/o servicio para establecer una línea base de los requisitos que sirva de base para definir los productos de trabajo necesarios.
- ENG.2 Análisis de requisitos del sistema: para transformar los requisitos definidos por el cliente en un conjunto de requisitos técnicos deseados que guiarán el diseño del sistema.
- ENG.4 Análisis de requisitos del software: para establecer los requisitos de los elementos software del sistema.

La gestión de requisitos es una parte importante de una organización que desarrolla sus propios proyectos de software, puesto que es vital para reducir los riesgos inherentes a ellos.

Las prácticas de gestión efectiva de los requerimientos aseguran que únicamente los requisitos necesarios estén disponibles a todos los miembros del equipo del proyecto y que los cambios sean realizados bajo circunstancias controladas. Los requisitos que son acordados y aprobados por todos los stakeholders del proyecto, contribuirán a garantizar que los objetivos establecidos se cumplan.

Para esto, las prácticas de gestión recomendadas para cada una de las actividades de la Ingeniería de Requisitos son:

- En la fase de especificación y modelamiento: implica comprender las necesidades de los stakeholders, los requisitos modelados y agruparlos en un repositorio o base de datos.
- Priorización: esta actividad asiste a los Gerentes de Proyectos en la resolución de conflictos, relacionados con determinar la importancia de los requisitos determinados.
- Negociación de los requisitos: la Ingeniería de Requisitos es esencialmente un proceso complejo de comunicación y negociación que involucra a clientes, Gerentes de Proyectos y Administradores. En muchas situaciones el conflicto es inherente en los requisitos, por lo que hay necesidad de negociar entre los stakeholders.

Ingeniería de sistemas integrada basada en modelos. Podrá agilizar los plazos y reducir los riesgos estableciendo un modelo del producto que defina todas las características arquitectónicas, funcionales y lógicas, así como los objetos de interfaz, conexión y soporte que rellenan automáticamente la definición del sistema y todas las condiciones variables. Podrá vincular los requisitos del sistema a definiciones de diagramas y de objetos, incorporando la trazabilidad necesaria para evaluar las repercusiones que tendrán los cambios en las siguientes fases.

El análisis y verificación del sistema. Podrá modelar, simular, analizar y optimizar el rendimiento crítico del sistema y la interacción con los subsistemas antes de generar

prototipos físicos, con herramientas para el modelado matemático, el cálculo y análisis de sistemas, podrá identificar y corregir antes los problemas de diseño del sistema o de interacción de los subsistemas. A medida que los modelos evolucionan, se gestiona las relaciones con el diseño para que pueda evaluar la repercusión de los cambios en las siguientes fases.

Ventajas

- Reducción de los riesgos al comprender los requisitos del producto, las restricciones del programa, las inquietudes de ingeniería y los problemas de la cadena de suministro y fabricación.
- Combinación del conocimiento al nivel de sistemas con el conocimiento de los requisitos para tomar decisiones mejores.
- Garantía de que se está lanzando contenido de producto adecuado en el momento adecuado y al mercado adecuado.
- Simplificación del bucle cerrado de información al reconocer cuándo está en riesgo la intención y comunicarlo a los involucrados en la gestión del ciclo de vida

CMMI

CMMI (Modelo de Madurez de Capacidad Integrado) pertenece a la familia de modelos desarrollados por el SEI (Software Engineering Institute) para evaluar las capacidades de las organizaciones de ingeniería de sistemas, ingeniería de software, además del desarrollo integrado del producto y del proceso. (Vates S.A, 2013)

Es un modelo normativo donde las prácticas detalladas caracterizan los tipos normales de comportamiento esperables en una organización que ejecuta proyectos a gran escala. La mejora continua de los procesos se basa en muchos pasos pequeños y evolutivos en vez de innovaciones revolucionarias.

CMMI proporciona un marco para organizar estos pasos evolutivos dentro de cinco niveles de maduración que sientan fundamentos sucesivos para la mejora continua del proceso. Los cuales se muestran en la ilustración 2,



Ilustración 3: Niveles de Madurez por CMMI

Niveles de madurez

Un Nivel de Madurez es una plataforma evolutiva bien definida destinada a lograr un proceso de software maduro. Cada nivel de madurez proporciona una capa en los cimientos para un proceso de mejora continua.

Cada nivel comprende un conjunto de objetivos que, una vez alcanzados, estabilizan un componente importante del proceso de software. Al alcanzar cada nivel del marco de madurez se establece un componente diferente en el proceso de software, resultando en un incremento en la capacidad de proceso de la organización.

Evaluación por CMMI

Hoy en día las compañías que producen software para satisfacer a un mercado creciente que reclama este tipo de soluciones tecnológicas deben buscar continuamente alternativas que les permitan mejorar su performance y calidad de productos para poder seguir compitiendo en un escenario cada vez más globalizado y agresivo.

Esta situación se torna más obvia cuando analizamos las necesidades de la producción global de software:

- Mejor performance
- Más eficiencia
- Evitar pérdidas de mercado

- Recursos humanos mejor preparados
- Productos que faciliten la integración de diferentes tecnologías.

Si visualizamos estos puntos de una manera más general podríamos afirmar que la industria tiene una fuerte necesidad de optimización de todas las facetas del desarrollo de productos. (Vates S.A, 2013).

2.2.5. NIVELES DE ORGANIZACIÓN

Existe una manera particular de trabajar, pero en las Tecnologías de la Información, debemos ser capaces de adaptarnos a estándares que nos permiten desempeñarnos en cualquier ambiente y en cualquier industria. Para ello existen certificaciones como TSP y PSP.

2.2.6.1. TSP (Team Software Process)

Team Software Process (TSP) es un método de establecimiento y mejora del trabajo en equipo para procesos software.

TSP proporciona directrices para ayudar a un equipo a establecer sus objetivos, a planificar sus procesos y a revisar su trabajo con el fin de que la organización pueda establecer prácticas de ingeniería avanzadas y así obtener productos eficientes, fiables y de calidad. Está formado por dos componentes primarios que abarcan distintos aspectos del trabajo en equipo: (WordPress, 2010).

- Formación del equipo de trabajo.
- Gestión del equipo de trabajo.

Maneja características tales como,

- Miembros expertos en papeles de liderazgo y pertenencia.
- Relaciones tranquilas y establecidas entre los miembros.
- Los miembros se sienten atraídos por el grupo y son fieles.
- Los miembros son flexibles y adaptables en sus metas y actitudes.

2.2.5.2.PSP (Personal Software Process)

El Personal Software Process, conocido por sus siglas como PSP, es una metodología de reciente creación, proveniente del Instituto de Ingeniería del Software (SEI). PSP es una alternativa dirigida a los ingenieros de sistemas, que les permite mejorar la forma en la que construyen software. Considerando aspectos como la planeación, calidad, estimación de costos y productividad.

Se citan algunas características

- PSP es una metodología basada en estimación. La estimación permite saber cuándo y cómo se desarrollan las tareas de un proceso, por lo que podría citarse como un aspecto importante de esta metodología el estar basada en métricas y estimaciones.
- permitirán al ingeniero de software identificar, tanto sus fortalezas como sus debilidades, y crecer a través de un proceso de auto aprendizaje y auto mejora.

3.3. ANTECEDENTES

Para asegurar la calidad de los procesos de desarrollo de software en una empresa, es necesario además de realizar mediciones en cuanto a los productos de software como tal, evaluar los procesos que componen el ciclo de vida de su desarrollo, debido a que se ha demostrado que la mayoría de mediciones se realizan sobre productos, y son escasas las mediciones sobre los procesos, por ende para que dicha evaluación sea satisfactoria es necesario que las empresas mantengan sus procesos gestionados (Gómez, Oktaba, García, & Piattini, 2010).

“La gestión de procesos de software identifica cuatro responsabilidades clave que son (I) definir el proceso, (II) medir el proceso, (III) controlar el proceso, y (IV) mejorar el proceso” (Florac, Park, & Carleton, 1997), cuando una empresa mantiene sus procesos gestionados, fortalece el nivel de calidad del software que desarrolla. Sin embargo “en muchas ocasiones estas mejoras se miden a través de procesos informales y subjetivos basados en la percepción de los empleados y/o auditores” (Pino, García, Serrano, & Piattini, 2006) por esta razón la gestión debe ser una operación formal y rigurosa, donde la empresa pueda tener avances y progresos realmente significativos mediante los cuales se pueda determinar el nivel de madurez de la organización con base en la capacidad de los procesos.

Actualmente existen distintos modelos y métodos, como CMM o CMMI y la norma ISO/IEC 12207:2008, que permiten evaluar los niveles de madurez en los procesos, con base en estos modelos y principalmente a la norma ISO/IEC 12207:1995 surge la norma ISO/IEC 15504:2003 que “inicialmente se orientó únicamente a evaluación de procesos software” (Basurto, 2009) pero en la actualidad puede ser aplicada para la evaluación de cualquier tipo de proceso, esta norma se caracteriza por “ser un marco para métodos de evaluación y no un método o modelo en sí” (Rodríguez, Alonso, & Sánchez, 2005). Sin embargo “la implantación en PYMEs de los modelos referentes en la actualidad, CMMI e ISO/IEC 15504, supone una gran inversión en tiempo dinero y recursos” (Garzás, Fernández, & Piattini, 2009), gracias a dicha implantación las empresas pueden resultar más competitivas, generar mayores utilidades y ser más atractivas a los clientes.

En el caso de Colombia “el proyecto Sistema Integral para el mejoramiento de los Procesos de Desarrollo de Software en Colombia, SIMEP-SW1 busca proporcionar a las empresas del sector informático de Colombia las herramientas necesarias para motivarlas a mejorar sus procesos de desarrollo de software” (Pino, García, Ruiz, & Piattini, 2006). Además, busca implementar un modelo ligero, como Light MECPDS, que principalmente toma la norma ISO/IEC15504:2003, basándose en tan solo tres niveles de capacidad (Nivel 0 “Proceso Incompleto”, Nivel 1 “Proceso Realizado”, Nivel 2 “Proceso Gestionado”) y ofreciendo un marco de trabajo para la medición y evaluación de los procesos del ciclo de vida.

Light MECPDS está definido con base a las necesidades de las empresas del área de informática de Colombia, sin embargo, este modelo es útil para cualquier PYME que haga parte de la industria de software, se pretende así, que Light MECPDS pueda implementarse en SITIS⁴ y otras PYMEs de Colombia.

⁴ SITIS: Soluciones Informáticas Integrales.

4. OBJETIVOS

4.1. OBJETIVO GENERAL

Desarrollar un software para la gestión de requisitos en los niveles 1 y 2 de madurez de la norma ISO/IEC 15504 (SPICE) en la empresa SOLINCES S.A.S., utilizando la metodología de desarrollo estándar RUP (Rational Unified Process).

4.2. OBJETIVOS ESPECÍFICOS

- ✓ Recopilar y analizar los procedimientos relacionados con el manejo y control de requisitos de la empresa SOLINCES S.A.S, a través de técnicas para la recolección de información y recomendaciones de suministro, definición y análisis de la norma ISO 15504 orientado al cumplimiento hasta el nivel 2 de la misma.
- ✓ Diseñar módulos por niveles de usuario, mediante el uso de herramientas de modelado, patrones de diseño y proceso de gestión de modelo de ciclo de vida de la norma ISO 15504 para comprender el sistema y organizar el desarrollo del mismo enfatizando los niveles 1 y 2 de madurez.
- ✓ Implementar la codificación del sistema mediante el uso de las buenas prácticas de programación para obtener un producto ejecutable que se pueda someter a pruebas.
- ✓ Evaluar el producto haciendo pruebas unitarias y de laboratorios para determinar el grado de cumplimiento con los requisitos iniciales.
- ✓ Documentar el desarrollo de la investigación en sus diferentes fases.

5. METODOLOGÍA

Al realizar los objetivos específicos de este proyecto, conllevó al cumplimiento del objetivo general, por esta razón se realizó paso a paso todas las actividades que conforman cada uno de ellos; se tomó como referente la metodología de desarrollo estándar RUP (Rational Unified Process), la cual consta de cuatro fases y de un flujo de trabajo, las actividades pueden ser concurrentes alternando la arquitectura con la elaboración del código fuente (Huamatuco, Vargas, & Quispe, 2011), en la Ilustración 5, se puede apreciar cómo está estructurada la metodología utilizada.

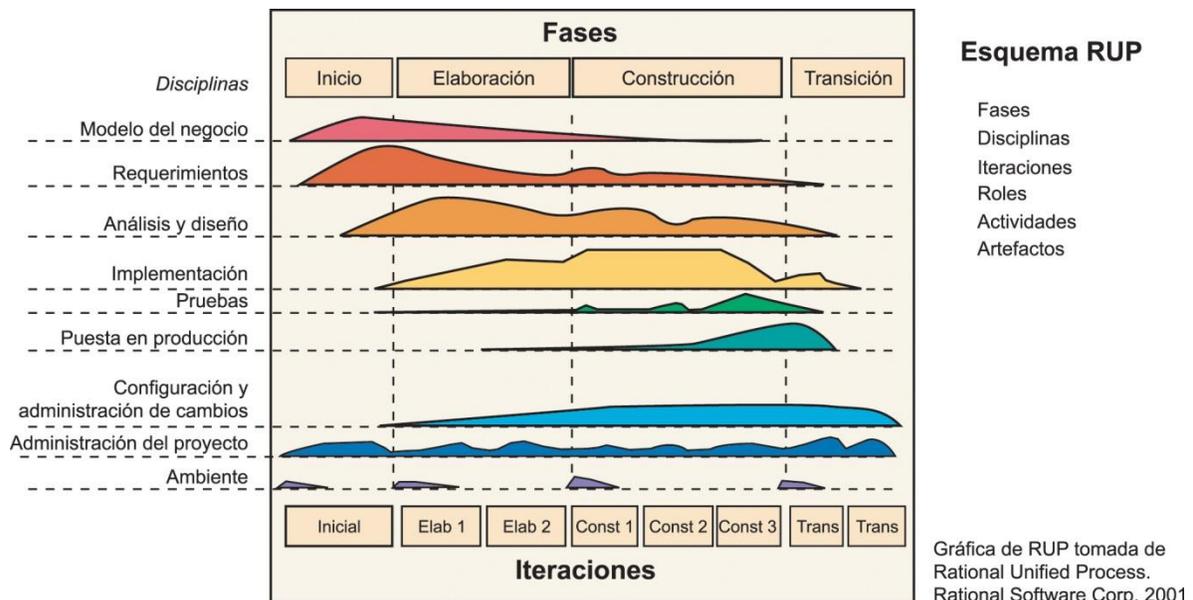


Ilustración 4: Estructura de la metodología RUP

5.1. TIPO DE INVESTIGACIÓN

Acorde a la metodología planteada, la investigación se considera de tipo aplicada debido a que se está determinando el comportamiento y las características de los procesos de ingeniería de requisitos en el desarrollo de software, el enfoque será cualitativo porque se medirá el nivel de satisfacción y cumplimiento de la norma en el área de la gestión de requisitos utilizando una norma con tal pertinencia como la norma internacional ISO/IEC 15504(SPICE).

Durante esta investigación se fijaron los objetivos específicos y se seleccionaron fuentes de información secundarias, con la colaboración y apoyo de los profesores del programa de Ingeniería de Sistemas de la Universidad de Cartagena, con el personal que desarrolla software en la empresa SOLINCES S.A.S. y con fuentes bibliográficas, tales como: libros, revistas y las bases de datos existentes en la biblioteca de la Universidad de Cartagena (ACS Publications, ASCE Library Ebsco Host e IEEE Computer Society Digital Library).

La investigación fue realizada en la ciudad de Cartagena de Indias en la empresa SOLINCES S.A.S., por un periodo aproximado de 8 meses comprendido desde febrero hasta septiembre de 2014, donde por medio de entrevistas, observación y encuestas se logró la recolección y análisis de los datos que permitieron desarrollar el proyecto a cabalidad.

5.2. FASES METODOLOGÍA RUP

Cada objetivo específico fue dividido en fases, las cuales describen el procedimiento que se realizaron en cada una de ellas de la siguiente manera:

5.2.1. FASE DE INICIO

Recopilar y analizar los procedimientos relacionados con el manejo y control de requisitos de la empresa SOLINCES S.A.S, a través de técnicas para la recolección de información y recomendaciones de suministro, definición y análisis de la norma ISO 15504 orientado al cumplimiento hasta el nivel 2 de la misma.

En esta fase de desarrollo se realizaron encuesta de satisfacción, donde el ingeniero Giraldo Turizo, representante legal de SOLINCES S.A.S., estipuló el manejo de los procesos en la empresa y con ello se realizó una medición orientada a un diagnóstico cuantitativo y/o cualitativo de variables relacionadas con el modelo SPICE en los niveles 1 y 2 aplicada a los procesos de ingeniería, gestión, relación cliente-proveedor de la organización y del soporte. Al utilizar este marco de referencia en la empresa SOLINCES S.A.S. se determinan las fortalezas y debilidades en sus procesos de compra, suministro, desarrollo, operación, mantenimiento y soporte del software.

Para determinar las variables se tuvo en cuenta *la Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504⁵*, donde se recopilan y analizan los procedimientos relacionados con el manejo y control de requisitos de la empresa SOLINCES S.A.S.

En la encuesta se evaluaron los siguientes ítems:

- Proporcionar al cliente un producto o servicio que cumpla con los requisitos acordados.
- Definir los requisitos del sistema para proporcionar los servicios necesarios a usuarios y otros afectados en un entorno definido.
- Trasformar los requisitos de los stakeholders en un conjunto deseado de requisitos técnicos del sistema que guiarán el diseño del sistema.
- Definir, mantener y asegurar la disponibilidad de políticas, procesos y modelos del ciclo de vida, para que sean utilizados por la organización.
- Elaborar y comunicar los planes de proyectos, de forma efectiva y viable.
- Determinar el estado del proyecto y asegurar que se realiza de acuerdo con los planes y el calendarios establecido, presupuestos planificados y satisfaciendo los objetivos técnicos.
- Establecer y mantener la integridad de los elementos que toman el producto software de un proceso o proyecto y ponerlos a disposición de las partes interesadas.
- Establecer y mantener la integridad de todos los productos de trabajo identificados de un proyecto a disposición de las partes interesadas.
- Recoger, analizar e informar sobre los datos relativos a los productos desarrollo y procesos implementados dentro de la unidad organizacional, para apoyar una gestión efectiva de los procesos y demostrar objetivamente la calidad de los productos.

⁵ Artículo de revisión orientado a empresas desarrolladoras de software, desarrollado en el Grupo de Investigación en Software – GIS, dicho grupo se encuentra avalado por la Universidad Pedagógica y Tecnológica de Colombia y Catalogado en COLCIENCIAS en categoría C. El desarrollo se llevó a cabo durante el segundo semestre de 2010.

- Asegurar que los productos de trabajo y los procesos cumplen con las disposiciones y planes predefinidos.

Por medio de los ítems antes mencionados se procedió a realizar un diagnóstico para identificar la planeación de los proyectos efectuados en SOLINCES S.A.S., a través de la siguiente tabla.

Tabla 9: Planeación del Proyecto

Planeación del Proyecto			
1	Revisar con el responsable de Gestión de Proyectos, la Descripción del Proyecto		
1.1	¿Existe un Responsable de la Gestión de Proyectos?	Si No	1.2 1.3
1.2	¿Quién Ocupa?	Rol:	1.3
1.3	¿Existes un Documento donde se definan objetivos y alcances del proyecto?	Si No	1.5 1.4
1.4	Si no existe el DP, ¿en que se basa para definir las actividades para cada proyecto?		1.5
1.5	¿Responde a una plantilla predefinida?	Si/No	1.6
1.6	¿En qué medio queda documentado? (manual/digital)	Manual/Digital	1.2

Con la anterior tabla se identifican las responsabilidades claves para la planeación de los proyectos.

Por otra parte la norma ISO/IEC 15504 para determinar si un proceso cumple un determinado nivel de capacidad, calcula el porcentaje de cumplimiento basándose en la siguiente tabla.

Tabla 10: Porcentaje de Cumplimiento basado en la Norma ISO/IEC 15504

Valores posibles del nivel		Grado de Cumplimiento
N	No alcanzado	0% - 15%
P	Parcialmente alcanzado	16% - 50%
L	Ampliamente alcanzada	51% - 81%
8F	Completamente alcanzado	86% - 100%

Fuente: (Revista Virtual Universidad Católica del Norte. No. 34, 2011)

A continuación se describen los valores de cada nivel:

N: Indica una poca o nula evidencia de que se ha alcanzado esta actividad en el proceso evaluado.

P: Se evidencia una aproximación sistemática del alcance del proceso pero algunas de sus características no se dan.

L: Hay bastantes evidencias de que se alcanza el atributo, pero la realización del proceso diverge en alguna área.

F: Hay evidencia de que el atributo se alcanza plenamente y de manera sistemática en el proceso evaluado y no hay debilidades importantes en la unidad organizacional en la que se ubica el proceso.

Los formatos aplicados para estas encuestas y determinar las variables se pueden observar en el **Anexo A y Anexo B**⁶ el cual permite identificar el alcance de esta basándose en los niveles 1 y 2 de la norma ISO 15504 definiendo los propósitos y los alcances del proyecto. (Alarcón Aldana, González Sanabria, & Rodríguez Torres, Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504, 2011).

Una vez identificadas necesidades básicas y los responsables se definieron una serie de actividades y/o tareas indispensables para cumplir las fases que nos brinda la metodología para el desarrollo de un software para la gestión de requisitos en SOLINCES S.A.S., cumpliéndose el objetivo general y el objetivo específico número uno. Dando paso a la implementación del objetivo número dos y a la fase de elaboración de la metodología RUP.

5.2.2. FASE DE ELABORACIÓN

Diseñar módulos por niveles de usuario, mediante el uso de herramientas de modelado, patrones de diseño y proceso de gestión de modelo de ciclo de vida de la norma ISO 15504 para comprender el sistema y organizar el desarrollo del mismo enfatizando los niveles 1 y 2 de madurez. El desarrollo de Software es un proceso complejo y a menudo difícil que requiere la síntesis de muchos sistemas. Desde el modelado y diseño hasta el código, administración del proyecto, pruebas, despliegue, administración de cambios y más.

En esta fase la realización de los modelos ayudó a visualizar el sistema de una manera que se estos obtuvieran un nivel de abstracción mayor, el cual puede utilizarse para la comunicación con el cliente a través de los diagramas UML, como diagramas de casos de uso, los cuales describen cómo un usuario interactuará con el software, diagramas de clase, lo cual modela las relaciones entre los objetos en el código y varios otros tipos de diagramas. El modelado con UML permite que los profesionales de la ingeniería del software entiendan completamente un diseño abstracto antes de adentrarse en la programación. Este proceso

⁶ Ver Anexos A y Anexo B en formato digital aplicado a la empresa SOLINCES S.A.S. en la carpeta de anexos.

también permite que una implementación sea validada y entendida después de que ha sido programada. (Sacha Moufarrege)

Se obtuvo como resultado un componente arquitectónico conformado por los modelos de caso de uso, de diseño, de análisis, de implementación y de prueba realizado con la ayuda del estándar StarUML.

5.2.3. FASE DE CONSTRUCCIÓN

Implementar la codificación del sistema mediante el uso de las buenas prácticas de programación para obtener un producto ejecutable que se pueda someter a pruebas. Para esta solución se aplicó los conocimientos adquiridos durante el transcurso de la carrera Ingeniería de Sistemas en asignaturas de la línea de desarrollo de software que van desde: algoritmos y programación básica hasta tópicos avanzados de ingeniería del software, haciendo especial énfasis en las asignaturas de ingeniería de servicios de internet, ingeniería del software, gestión de la calidad prueba y mantenimiento de software y gestión de proyectos de ingeniería del software.

En primera instancia y partiendo de los modelados planteados en la fase de elaboración se realizó una versión del software la cual fue presentada el 26 de junio de 2014, con el ingeniero Giraldo Turizo Cano, Representante Legal de SOLINCES S.A.S y con la ingeniera jefe de proyectos Katherine Marrugo, *ver Anexo C.* en donde se estipularon algunas recomendaciones para mayor navegabilidad del software y con ello realizar una segunda versión del software final para posteriormente someterlo a pruebas.

5.2.4. FASE DE TRANSICIÓN

Una vez presentado el producto (software) final del proyecto, se inicia la fase de transición, en la cual se asegura que el software esté disponible para los usuarios finales, y con ello se ejecutar el último objetivo específico.

Evaluar el producto haciendo pruebas unitarias y de laboratorios para determinar el grado de cumplimiento con los requisitos iniciales. Se realizaron pruebas para medir el grado de

cumplimiento del software en los requisitos iniciales, realizando pruebas unitarias y simuladas acorde a lo estipulado en el diseño y plan de prueba.

Para ello, se realizó un plan y diseño de pruebas con el propósito de establecer las técnicas, herramientas y actividades relacionadas con la ejecución y validación de pruebas; incluye responsabilidades de cada una de las tareas, los recursos y los prerrequisitos que deben ser considerados en el esfuerzo de cada una de los módulos, permitiendo garantizar el cumplimiento de los requerimientos planteados en el marco del desarrollo del proyecto denominado “SOFTWARE PARA LA GESTIÓN DE REQUISITOS EN LOS NIVELES 1 Y 2 DE LA NORMA ISO/IEC 15504 (SPICE) EN LA EMPRESA SOLINCES S.A.S”. *Ver anexo D – FORMATO DISEÑO Y PLAN DE PRUEBAS.*

6. RESULTADOS Y DISCUSIÓN

A continuación se presentan los resultados arrojados en el transcurso de la investigación. Dichos resultados van relacionados con los objetivos específicos planteados y se describen a continuación:

6.1. RESULTADOS DEL OBJETIVO 1

Recopilar y analizar los procedimientos relacionados con el manejo y control de requisitos de la empresa SOLINCES S.A.S, a través de técnicas para la recolección de información y recomendaciones de suministro, definición y análisis de la norma ISO 15504 orientado al cumplimiento hasta el nivel 2 de la misma.

Según la información presentada en el portal de la norma en español, en su aparte de niveles de madurez, “en el nivel 1 la organización simplemente implementa y alcanza de manera básica los resultados del proceso” (INTECO, 2008), y al alcanzar los resultados propuestos es posible identificar satisfactoriamente las salidas (resultados) del proceso evaluado.

Para el nivel de madurez 2 (organización gestionada) adopta los mismos procesos del nivel de madurez 1, con la diferencia que “en el nivel 2 la organización además de implementar los objetivos de los procesos, demuestra una planificación, seguimiento y control tanto de los procesos como de sus productos de trabajo asociados” (INTECO, 2008).

Para realizar la recopilación y análisis en la empresa SOLINCES S.A.S. se realizó lo siguiente:

➤ **Encuesta**

Para la creación del software se realizó una encuesta al Ingeniero Giraldo Turizo Cano, Representante Legal de SOLINCES S.A.S. la cual permito obtener información como:

- ✓ Las responsabilidades claves para la gestión de la realización de los procesos.
- ✓ Herramientas informáticas utilizadas como apoyo a la gestión de los proyectos software.

- ✓ Identificar los resultados de la implementación de los objetivos de los procesos que alcanza satisfactoriamente SOLINCES S.A.S. Teniendo en cuenta los procesos básicos en el nivel 1 de madurez de la norma ISO 15504.
- ✓ Identificar los procesos de planificación, seguimiento y control tanto de los procesos como de sus productos de trabajo asociados por SOLINCES S.A.S. Teniendo en cuenta los procesos básicos en el nivel 2 de madurez de la norma ISO 15504.

Para ello se grabó el audio de la encuesta en la que también participó el profesor Plinio Puello Marrugo (tutor del proyecto de grado), este audio se encuentra en la carpeta de Anexos con el nombre de *Audio_Encuesta 1*.

Una vez obtenidos los procedimientos relacionados con el manejo y control de requisitos de la empresa se procedió a realizar un análisis teniendo en cuenta las recomendaciones de suministro, definición y análisis de la norma ISO/IEC 15504 orientado al cumplimiento hasta el nivel 2 de la misma.

Análisis del Anexo A

Teniendo en cuenta el anexo A, se realiza el siguiente análisis:

En cualquier proyecto de desarrollo de software, es necesario disponer de una gestión eficaz para garantizar que el sistema cumpla los objetivos y que se desarrolle dentro de un presupuesto aceptable.

Al conocer que la herramienta informática implementada por la empresa SOLINCE S.A.S. para garantizar que el sistema cumpla los objetivos y que se desarrolle dentro de un presupuesto aceptable es Microsoft Project⁷, se realiza una comparación teniendo en cuenta el beneficio que esta le ofrece y los esperados por la norma ISO/IEC 15504.

El análisis de este proceso se ve reflejado en la ilustración 5 donde el indicador de resultados 1 (uno) se toma como respuesta SI y el indicador de resultados 0 (cero) se toma como respuesta NO.

⁷ Software de administración de proyectos diseñado, 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.

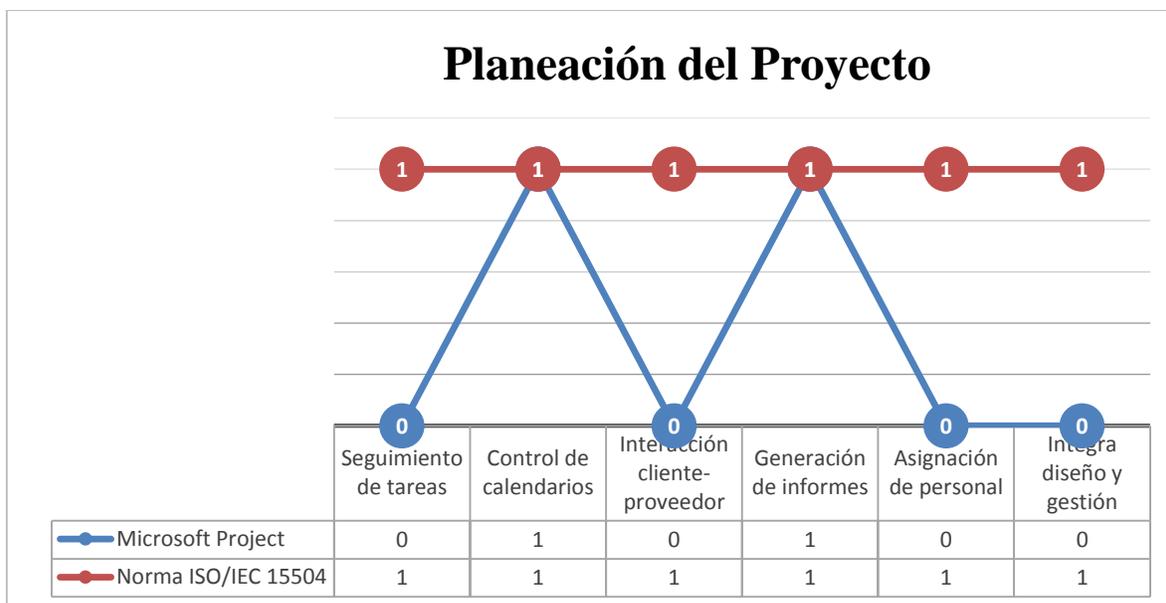


Ilustración 5: Análisis de Planeación del Proyecto. Fuente (Autores)

De la ilustración anterior, en comparación con la norma ISO/IEC 15504 y Microsoft Project, esta última herramienta implementada por la empresa SOLINCES S.A.S. se denota que esta cumple o cuenta con dos actividades o beneficios (control de calendario y generación de informe) para la planeación de proyecto software, la empresa al implementar la norma puede contar con los beneficios para la implementación de un buen proyecto de software como son seguimientos de tareas, interacción cliente-proveedor, asignación de personal, integración diseño y gestión, control de calendario y generación de informe.

Análisis del Anexo B

Teniendo en cuenta el anexo B, se realiza el siguiente análisis:

Procesos:

Suministro (SUM): Proporcionar al cliente un producto o servicio que cumpla con los requisitos acordados.

El análisis de este proceso se ve reflejado en la ilustración 5 donde el indicador de resultados 1 (uno) se toma como respuesta SI y el indicador de resultados 0 (cero) se toma como respuesta NO.

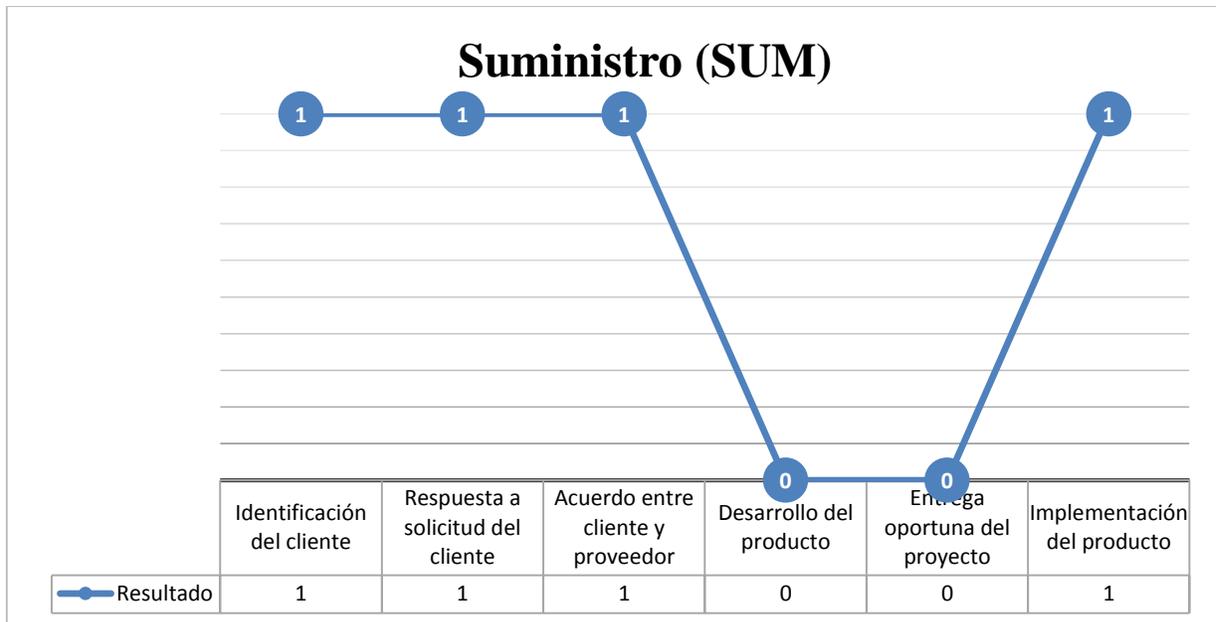


Ilustración 6: Análisis de Suministro. Fuente (Autores)

La ilustración anterior muestra que la empresa cumple satisfactoriamente los procesos de identificación de cliente, respuesta a solicitud de cliente, acuerdo entre cliente y proveedor y la implementación del producto pero presenta falencia con el desarrollo del producto y con la entrega oportuna del proyecto que se encuentra realizando.

Con la ilustración 6 se concluye teniendo en cuenta los procesos implementados por la norma ISO/IEC 15505 que la empresa realiza un 66% en el proceso de suministro.

Definición de los Requisitos de Usuarios (RQSI): Definir los requisitos del sistema para proporcionar los servicios necesarios a usuarios y otros afectados en un entorno definido.

El análisis de este proceso se ve reflejado en la ilustración 6 donde el indicador de resultados 1 (uno) se toma como respuesta SI y el indicador de resultados 0 (cero) se toma como respuesta NO.

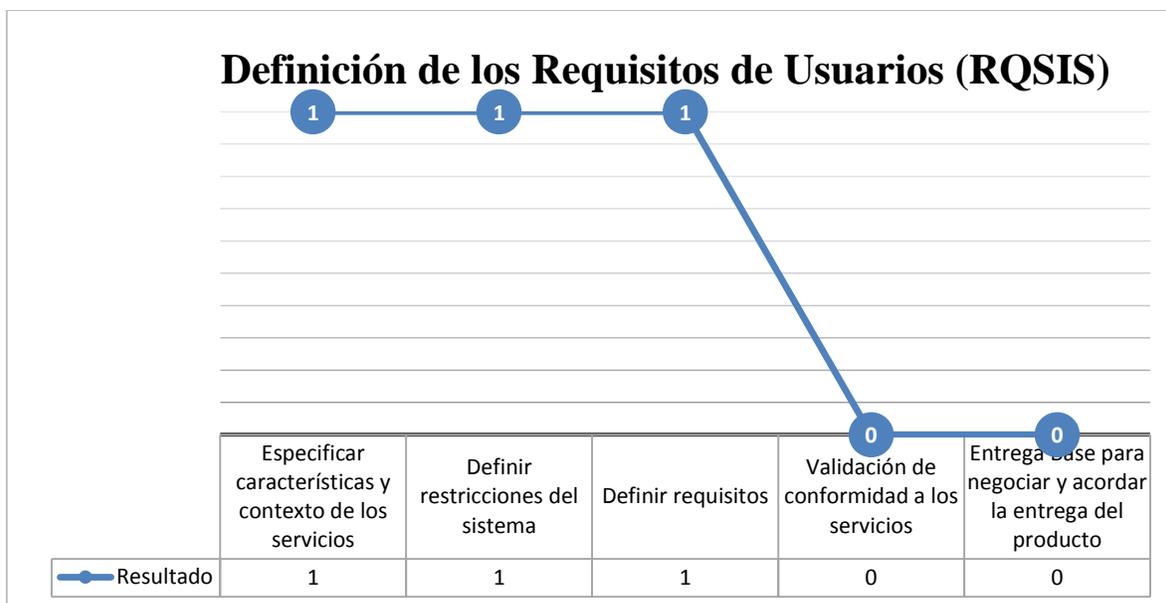


Ilustración 7: Análisis Definición de los Requisitos de Usuarios. Fuente (Autores)

La ilustración anterior muestra que la empresa cumple satisfactoriamente los procesos de especificar características y contexto de los servicios, definir restricciones del sistema, definir requisitos pero presenta falencia con la validación de conformidad a los servicios y base para negociar y acordar la entrega del producto.

Con la ilustración 7 se concluye teniendo en cuenta los procesos implementados por la norma ISO/IEC 15505 que la empresa realiza un 60% en el proceso de definición de los requisitos de usuarios.

Análisis de los Requisitos del Sistema (RQSI): Transformar los requisitos de los stakeholders en un conjunto deseado de requisitos técnicos del sistema que guiarán el diseño del sistema.

El análisis de este proceso se ve reflejado en la ilustración 7 donde el indicador de resultados 1 (uno) se toma como respuesta SI y el indicador de resultados 0 (cero) se toma como respuesta NO.

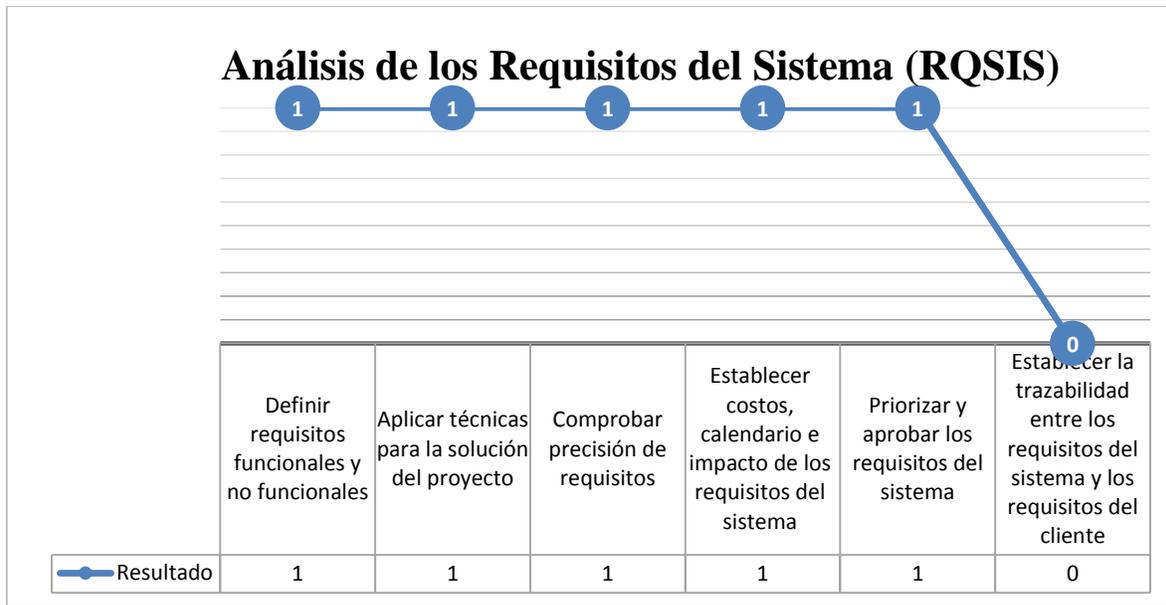


Ilustración 8: Análisis de los Requisitos del Sistema

La ilustración anterior muestra que la empresa cumple satisfactoriamente los procesos de definir requisitos funcionales y no funcionales, aplicar técnicas para la solución del proyecto, comprobar precisión de requisitos, establecer costos, calendario e impacto de los requisitos del sistema en el entorno de explotación, priorizar y aprobar los requisitos del sistema pero presenta falencia con establecer la trazabilidad entre los requisitos del sistema y los requisitos del cliente.

Con la ilustración 8 se concluye teniendo en cuenta los procesos implementados por la norma ISO/IEC 15505 que la empresa realiza un 83% en el proceso de definición de los requisitos de usuarios.

De este análisis se puede concluir que la implementación de la norma ISO/IEC 15504 en la empresa SOLINCES S.A.S., esta puede alcanzar una mejora en los procesos de organización basados en la gestión de requisitos. Lo anterior da cumplimiento al primero objetivo específico planteado para el desarrollo de esta investigación y así dar paso al siguiente objetivo.

6.2. RESULTADOS DEL OBJETIVO 2

Diseñar módulos por niveles de usuario, mediante el uso de herramientas de modelado, patrones de diseño y proceso de gestión de modelo de ciclo de vida de la norma ISO 15504 para comprender el sistema y organizar el desarrollo del mismo enfatizando los niveles 1 y 2 de madurez.

Para la aplicación del diseño utilizado se optó por el estándar StarUML, con el cual se describe el ensamble de los componentes del software, verificar que se comprenda correctamente el ambiente del usuario.

A continuación se ilustra la tabla 11 con los diferentes modelos implementados para la creación del software.

Tabla 11: Modelos de Software

Modelo de Negocio
Modelo del Dominio
Casos de uso del mundo real
Modelo de Diseño
Vista lógica
<i>Vista de comportamiento:</i>
Diagramas general de casos de uso
Diagramas de secuencias
Diseño de la interfaz gráfica de usuario
<i>Vista Estructural:</i>
Diagrama de clases
Diseño de clases
Modelo de Implementación
Vista de despliegue

Modelo Base de Datos

Para observar los diferentes modelos realizados se encuentran en la carpeta de Anexos: **Modelos de Software.**

Cada uno de los modelos realizados describe un enfoque diferente para diferentes actividades que tienen lugar durante el proceso, permitiendo la construcción de un sistema donde se define la estructura, el comportamiento y la funcionalidad del mismo.

6.3. RESULTADOS DEL OBJETIVO 3

Implementar la codificación del sistema mediante el uso de las buenas prácticas de programación para obtener un producto ejecutable que se pueda someter a pruebas.

El modelado de Casos de Uso es la técnica más efectiva y a la vez la más simple para modelar los requisitos del sistema desde la perspectiva del usuario. Los Casos de Uso se utilizan para modelar cómo un sistema o negocio funciona actualmente, o cómo los usuarios desean que funcione. No es realmente una aproximación a la orientación a objetos; es realmente una forma de modelar procesos. Es, sin embargo, una manera muy buena de dirigirse hacia el análisis de sistemas orientado a objetos. Los casos de uso son generalmente el punto de partida del análisis orientado a objetos con UML. (Popkin Software and Systems)

La ilustración 9 presenta el caso de uso general que permiten definir la arquitectura base del sistema y el comportamiento del software en los diferentes módulos implementados.

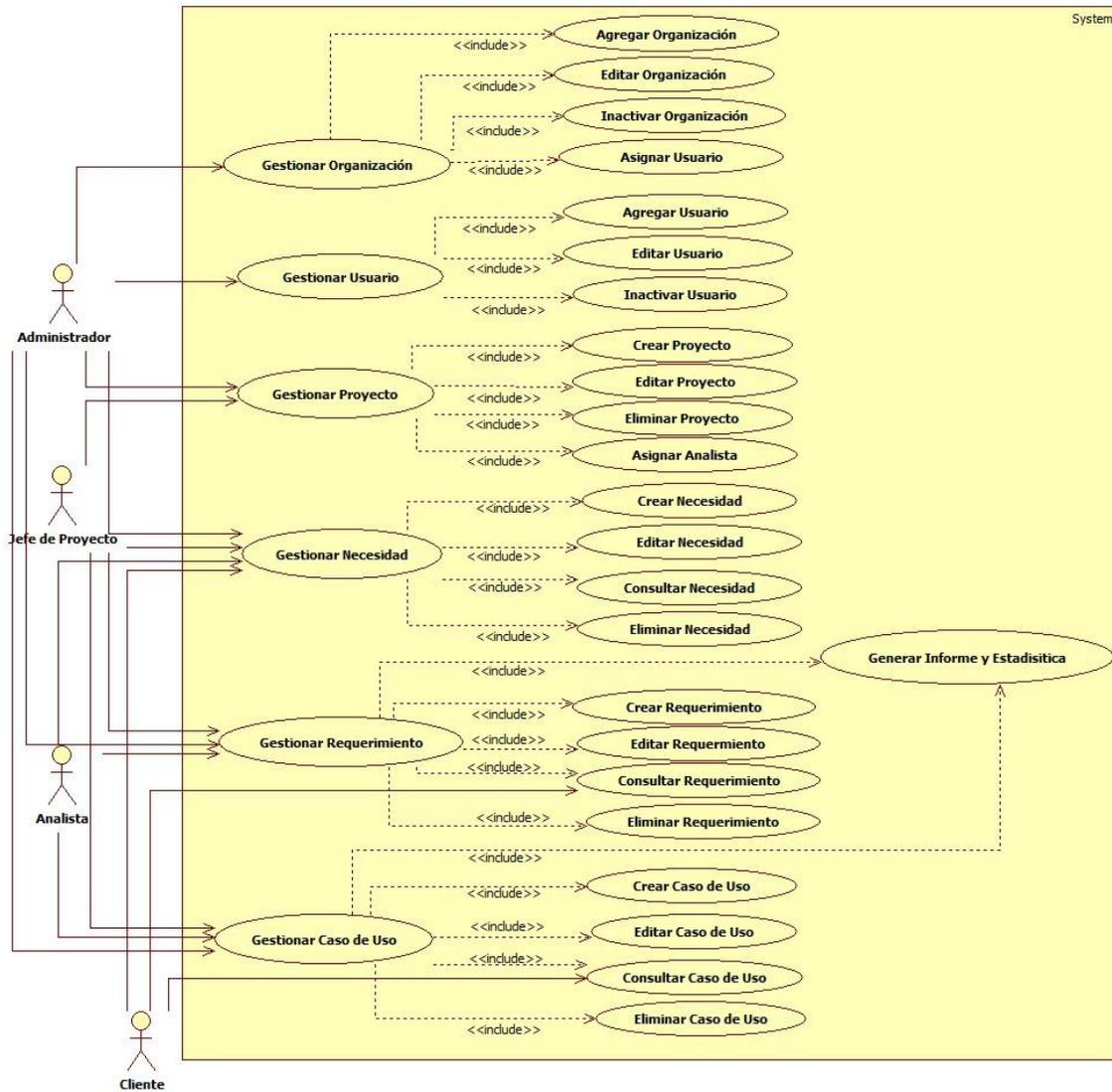


Ilustración 9: Caso de Uso General

A continuación se realizara la descripción de los casos de uso del sistema gestión de requisitos en los niveles 1 y 2 de la norma ISO 15504 (SPICE), el cual es uno de los diagramas más efectivos para modelar interacción entre objetos en un sistema. Un diagrama de caso de uso permite el modelado de una vista del escenario.

Descripción Caso de Uso

Tabla 12: Caso de Uso – Agregar Organización

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: ingresa los datos para el registro de la organización o empresa que solicita el servicio a SOLINCES S.A.S.
Precondiciones	Tener autorización para ingresar en el sistema.
Pos condición	Se registran los datos de formulario
Flujo normal	<ol style="list-style-type: none"> Administrador: Introduce usuario y contraseña. Sistema: valida los datos Administrador: ingresa datos de la organización o empresa Sistema: guarda el registro.
Flujo Alternativo	<p><i>El sistema no permite el acceso.</i></p> <ul style="list-style-type: none"> ✓ verificar usuario y contraseña correctamente. <p><i>El sistema no guarda los datos.</i></p> <ul style="list-style-type: none"> ✓ Verifique que el formulario no contenga datos no permitidos.
Frecuencia	Cada vez que se requiera registrar una organización o una empresa

Tabla 13: Caso de Uso - Editar Organización

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: Selección la organización que desea modificar
Precondiciones	La Organización debe estar registrada.
Pos condición	Administrador: Edita el registro de una organización.
Flujo normal	<ol style="list-style-type: none"> Administrador: selecciona organización que desea editar. Sistema: carga los datos de la organización a modificar. Administrador: ingresa nuevos datos de la organización. Sistema: guarda el registro.
Flujo Alternativo	<p><i>El sistema no permite la edición.</i></p> <ul style="list-style-type: none"> ✓ Verificar que la organización este activa. <p>El administrador no puede guardar el registro.</p> <ul style="list-style-type: none"> ✓ Verificar los datos del formulario, para que sean datos admitidos por éste.
Frecuencia	Cada vez que se requiera Editar la información de una organización

Tabla 14: Caso de Uso - Inactivar Organización

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: selección la organización que desea inactivar.
Precondiciones	La organización debe estar registrada
Pos condición	Administrador: desactiva la organización
Flujo normal	<ol style="list-style-type: none"> 1. Administrador: selecciona la organización que desea eliminar. 2. Sistema: solicita confirmación de la desactivación de la organización. 3. Administrador: desactiva organización 4. Sistema: confirma eliminado de registro.
Flujo Alternativo	<p><i>El sistema no desactiva la organización.</i></p> <p>✓ Verificar que la organización no tenga proyectos pendientes.</p>
Frecuencia	Cada vez que se requiera desactivar una organización

Tabla 15: Caso de Uso - Asignar Usuario

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: escoge la opción para asignar ingeniero jefe o analista o cliente
Precondiciones	Los usuarios deben de estar registrados
Pos condición	Administrador: usuario asignado a la organización
Flujo normal	<ol style="list-style-type: none"> 5. Administrador: seleccione la opción para asignar usuarios 6. Sistema: carga listado de usuarios 7. Administrador: selecciona el usuarios a asignar 8. Sistema: confirma asignación exitosa
Flujo Alternativo	<p><i>El sistema no asigna el usuario.</i></p> <p>✓ Verificar que allá usuarios registrados</p>
Frecuencia	Cada vez que se requiera asignar un usuario

Tabla 16: Caso de Uso - Agregar Usuario

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: ingresa los datos para el registro del usuario
Precondiciones	Tener autorización para ingresar en el sistema.
Pos condición	Se registran los datos de formulario
Flujo normal	<ol style="list-style-type: none"> 5. Administrador: Introduce usuario y contraseña. 6. Sistema: valida los datos 7. Administrador: ingresa datos del usuario 8. Sistema: guarda el registro de usuario
Flujo Alternativo	<p><i>El sistema no permite el acceso.</i></p> <ul style="list-style-type: none"> ✓ verificar usuario y contraseña correctamente. <p><i>El sistema no guarda los datos.</i></p> <ul style="list-style-type: none"> ✓ Verifique que el formulario no contenga datos no permitidos.
Frecuencia	Cada vez que se requiera registrar un usuario

Tabla 17: Caso de Uso - Editar Usuario

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: Selección el usuario que desea editar
Precondiciones	La Organización debe estar registrada.
Pos condición	Administrador: Edita el registro de un usuario.
Flujo normal	<ol style="list-style-type: none"> 5. Administrador: selecciona el registro del usuario que desea editar. 6. Sistema: carga los datos del usuario a editar. 7. Administrador: ingresa nuevos datos del usuario. 8. Sistema: guarda el registro.
Flujo Alternativo	<p><i>El sistema no permite la edición.</i></p> <ul style="list-style-type: none"> ✓ Verificar que el usuario este activa. <p>El administrador no puede guardar el registro.</p> <ul style="list-style-type: none"> ✓ Verificar los datos del formulario, para que sean datos admitidos por éste.
Frecuencia	Cada vez que se requiera Editar la información de una organización

Tabla 18: Caso de Uso - Inactivar Usuario

Actores	✓ Administrador
Personal involucrado e intereses	Administrador: selección el usuario que desea inactivar.
Precondiciones	El usuario debe estar registrado
Pos condición	Administrador: desactiva el usuario
Flujo normal	<p>9. Administrador: selecciona el usuario que desea eliminar.</p> <p>10. Sistema: solicita confirmación de la desactivación de el usuario.</p> <p>11. Administrador: desactivar usuario</p> <p>12. Sistema: confirma eliminado de registro.</p>
Flujo Alternativo	<p><i>El sistema no desactiva el usuario.</i></p> <p>✓ Verificar que el usuario no tenga proyectos pendientes.</p>
Frecuencia	Cada vez que se requiera desactivar una organización

Tabla 19: Caso de Uso - Crear Proyecto

Actores	✓ Administrador , Jefe de Proyecto
Personal involucrado e intereses	Administrador: crear un proyecto a la organización seleccionada Jefe de proyecto: crear un proyecto a la organización asignada
Precondiciones	La organización debe estar registrada
Pos condición	proyecto creado
Flujo normal	<p>13. Administrador, Jefe de Proyecto: selecciona la organización para crear proyecto nuevo</p> <p>14. Sistema: carga formulario para registro</p> <p>15. Administrador, Jefe de Proyecto: ingresan datos para el registro del proyecto</p> <p>16. Sistema: confirma la creación del proyecto</p>
Flujo Alternativo	<p><i>El sistema carga el formulario</i></p> <p>✓ Verificar que la organización este seleccionada.</p> <p>El administrador no puede guardar el registro.</p> <p>✓ Verificar los datos del formulario, para que sean datos admitidos por éste.</p>
Frecuencia	Cada vez que se requiera desactivar una organización

Tabla 20. Caso de Uso - Editar Proyecto

Actores	✓ Administrador, Jefe de Proyecto
Personal involucrado e intereses	Administrador: editar los datos del proyecto seleccionado Jefe de Proyecto: editar los datos del proyecto asignado
Precondiciones	Proyecto creado
Pos condición	Administrador, Jefe de Proyecto: Edita los datos del proyecto.
Flujo normal	<p>9. Administrador, Jefe de Proyecto: selecciona el proyecto de la organización que desea editar.</p> <p>10. Sistema: carga los datos del proyecto a editar.</p> <p>11. Administrador, Jefe de Proyecto: ingresa nuevos datos del proyecto seleccionado.</p> <p>12. Sistema: guarda el registro.</p>
Flujo Alternativo	<p><i>El sistema modifica los datos</i></p> <p>✓ Verificar que el proyecto este seleccionada.</p> <p>El administrador no puede guardar los cambios.</p> <p>✓ Verificar los datos del formulario, para que sean datos admitidos por éste.</p>
Frecuencia	Cada vez que se requiera Editar la información de un proyecto

Tabla 21: Caso de Uso - Eliminar Proyecto

Actores	✓ Administrador, Jefe de Proyecto
Personal involucrado e intereses	Administrador Jefe de Proyecto: selección el proyecto que desea eliminar
Precondiciones	El proyecto debe estar creado
Pos condición	Jefe de Proyecto: Eliminar el proyecto deseado
Flujo normal	<p>17. Administrador, Jefe de Proyecto: selecciona el proyecto que desea eliminar.</p> <p>18. Sistema: carga operación de eliminación</p> <p>19. Administrador, Jefe de Proyecto: eliminar el proyecto seleccionado</p> <p>20. Sistema: confirma eliminación del proyecto</p>
Flujo Alternativo	<p><i>El sistema no eliminar el proyecto</i></p> <p>✓ Verificar que el proyectos tenga actividades programadas</p>
Frecuencia	Cada vez que se requiera eliminar un proyecto

Tabla 22: Caso de Uso - Asignar Analista

Actores	✓ Administrador, Jefe de Proyecto
Personal involucrado e intereses	Administrador Jefe de Proyecto: asignar analista al proyecto seleccionado
Precondiciones	<ul style="list-style-type: none"> ✓ El proyecto y el analista debe estar creados ✓ El analista debe estar registrado en la misma organización del proyecto
Pos condición	Jefe de Proyecto: analista asignado satisfactoriamente
Flujo normal	<p>21. Administrador, Jefe de Proyecto: selecciona la organización al que desea asignar analistas.</p> <p>22. Sistema: carga los proyectos creados</p> <p>23. Administrador, Jefe de Proyecto: selección el proyecto para asignar analista</p> <p>24. Sistema: carga analista</p> <p>25. Administrador, Jefe de Proyecto: selecciona analista deseado</p> <p>26. Sistema: Carga analista</p>
Flujo Alternativo	<p><i>El sistema no asigna el analista</i></p> <ul style="list-style-type: none"> ✓ Verificar que el analista este activo
Frecuencia	Cada vez que se requiera asignar un analista a un proyecto

Tabla 23: Caso de Uso - Crear Necesidad

Actores	✓ Administrador, Jefe de Proyecto, Analista, Cliente
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista, Cliente: crear la necesidad satisfactoriamente
Precondiciones	✓ Tener un proyecto creado en la organización
Pos condición	Administrador, Jefe de Proyecto, Analista, Cliente: necesidad creada satisfactoriamente
Flujo normal	<p>27. Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar proyector.</p> <p>28. Sistema: carga los proyectos creados</p> <p>29. Administrador, Jefe de Proyecto, Analista, Cliente: diligenciar el formato para crear la necesidad</p> <p>30. Sistema: guardar necesidad</p>
Flujo Alternativo	<p><i>El sistema no crea la necesidad</i></p> <ul style="list-style-type: none"> ✓ Verificar los datos del formulario, para que sean datos admitidos por éste.
Frecuencia	Cada vez que se requiera asignar un analista a un proyecto

Tabla 24: Caso de Uso - Editar Necesidad

Actores	✓ Administrador, Jefe de Proyecto, Analista, Cliente
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista, Cliente: editar la necesidad satisfactoriamente
Precondiciones	✓ La necesidad debe estar creada
Pos condición	Administrador, Jefe de Proyecto, Analista, Cliente:: editar la necesidad satisfactoriamente
Flujo normal	<ol style="list-style-type: none"> 31. Administrador, Jefe de Proyecto, Analista, Cliente: selecciona la necesidad a editar 32. Sistema: carga la necesidad seleccionada 33. Administrador, Jefe de Proyecto, Analista, Cliente: ingresas nuevos datos para editar la necesidad 34. Sistema: necesidad editada satisfactoriamente
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <ul style="list-style-type: none"> ✓ Verificar que el registro exista. <p>Administrador, Jefe de Proyecto, Analista, Cliente no puede guardar el registro.</p> <ul style="list-style-type: none"> ✓ Verificar los datos del formulario, para que sean datos admitidos por éste o estén incompletos
Frecuencia	Cada vez que se requiera editar una necesidad

Tabla 25: Caso de Uso - Consultar Necesidad

Actores	✓ Administrador, Jefe de Proyecto, Analista, Cliente
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista, Cliente: escoge el proyecto al cual pertenece la necesidad
Precondiciones	La necesidad debe estar creada.
Pos condición	Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar ver necesidad
Flujo normal	<ol style="list-style-type: none"> 1. Administrador, Jefe de Proyecto, Analista, Cliente: escoge el proyecto al que pertenece la necesidad 2. Sistema: cargar los datos de ese proyecto. 3. Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar ver necesidad 4. Sistema: guarda el registro.
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <ul style="list-style-type: none"> ✓ Verificar que el registro exista.
Frecuencia	Cada vez que se requiera consultar una necesidad

Tabla 26: Casos de Uso - Eliminar Necesidad

Actores	✓ Administrador, Jefe de Proyecto, Analista, Cliente
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar la necesidad que desea eliminar.
Precondiciones	La necesidad debe estar registrado.
Pos condición	Administrador, Jefe de Proyecto, Analista, Cliente: eliminar la necesidad registrada
Flujo normal	<ol style="list-style-type: none"> 1. Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar la necesidad que desea eliminar. 2. Sistema: confirmar la eliminación de la necesidad 3. Administrador, Jefe de Proyecto, Analista, Cliente: elimina necesidad 4. Sistema: necesidad eliminada satisfactoriamente.
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <p>✓ Verificar que el registro exista.</p>
Frecuencia	Cada vez que se requiera eliminar una necesidad

Tabla 27: Caso de Uso - Crear Requerimiento

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: crear el requerimiento satisfactoriamente
Precondiciones	✓ Tener un proyecto creado en la organización
Pos condición	Administrador, Jefe de Proyecto, Analista: requerimiento creado satisfactoriamente
Flujo normal	<ol style="list-style-type: none"> 35. Administrador, Jefe de Proyecto, Analista: seleccionar proyector. 36. Sistema: carga registros 37. Administrador, Jefe de Proyecto, Analista: Seleccionar Necesidad 38. Sistema: Cargar necesidad 39. Administrador, Jefe de Proyecto, Analista: diligenciar el formato para crear el requerimiento 40. Sistema: guardar requerimiento
Flujo Alternativo	<p><i>El sistema no crea el requerimiento</i></p> <p>✓ Verificar los datos del formulario, para que sean datos admitidos por éste.</p>
Frecuencia	Cada vez que se requiera crear un requerimiento

Tabla 28: Caso de Uso - Editar Requerimiento

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: editar el requerimiento satisfactoriamente
Precondiciones	✓ El requerimiento debe estar creada
Pos condición	Administrador, Jefe de Proyecto, Analista: editar el requerimiento satisfactoriamente
Flujo normal	<p>41. Administrador, Jefe de Proyecto, Analista: seleccionar el requerimiento a editar</p> <p>42. Sistema: carga el requerimiento seleccionado</p> <p>43. Administrador, Jefe de Proyecto, Analista: ingresas nuevos datos para editar el requerimiento</p> <p>44. Sistema: requerimiento editado satisfactoriamente</p>
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <p>✓ Verificar que el registro exista.</p> <p>Administrador, Jefe de Proyecto, Analista no puede guardar el registro.</p> <p>✓ Verificar los datos del formulario, para que sean datos admitidos por éste o estén incompletos</p>
Frecuencia	Cada vez que se requiera editar un requerimiento

Tabla 29: Caso de Uso - Consultar Requerimiento

Actores	✓ Administrador, Jefe de Proyecto, Analista, Cliente
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista, Cliente: escoge el proyecto al cual pertenece el requerimiento
Precondiciones	La necesidad debe estar creada.
Pos condición	Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar ver requerimiento
Flujo normal	<p>5. Administrador, Jefe de Proyecto, Analista, Cliente: escoge el proyecto al que pertenece el requerimiento</p> <p>6. Sistema: cargar los datos de ese proyecto.</p> <p>7. Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar ver requerimiento</p> <p>8. Sistema: muestra el registro del requerimiento</p>
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <p>✓ Verificar que el registro exista.</p>
Frecuencia	Cada vez que se requiera consultar un requerimiento

Tabla 30: Caso de Uso - Eliminar Requerimiento

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: seleccionar el requerimiento que desea eliminar.
Precondiciones	El requerimiento debe estar registrado.
Pos condición	Administrador, Jefe de Proyecto, Analista: eliminar el requerimiento registrada
Flujo normal	<p>5. Administrador, Jefe de Proyecto, Analista: seleccionar el requerimiento que desea eliminar.</p> <p>6. Sistema: confirmar la eliminación del requerimiento</p> <p>7. Administrador, Jefe de Proyecto, Analista: elimina requerimiento</p> <p>8. Sistema: requerimiento eliminada satisfactoriamente.</p>
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <p>✓ Verificar que el registro exista.</p>
Frecuencia	Cada vez que se requiera eliminar un requerimiento

Tabla 31: Caso de Uso - Crear Caso de Uso

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: crear un caso de uso satisfactoriamente
Precondiciones	✓ Tener un proyecto creado en la organización
Pos condición	Administrador, Jefe de Proyecto, Analista: un caso de uso creado satisfactoriamente
Flujo normal	<p>45. Administrador, Jefe de Proyecto, Analista: seleccionar proyecto</p> <p>46. Sistema: carga los proyectos creados</p> <p>47. Administrador, Jefe de Proyecto, Analista: diligenciar el formato para crear un caso de uso</p> <p>48. Sistema: guardar un caso de uso</p>
Flujo Alternativo	<p><i>El sistema no crea caso de uso</i></p> <p>✓ Verificar los datos del formulario, para que sean datos admitidos por éste.</p>
Frecuencia	Cada vez que se requiera crear un caso de uso

Tabla 32: Caso de Uso - Editar Caso de Uso

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: editar un caso de uso satisfactoriamente
Precondiciones	✓ debe estar creada un caso de uso
Pos condición	Administrador, Jefe de Proyecto, Analista: editar un caso de uso satisfactoriamente
Flujo normal	<p>49. Administrador, Jefe de Proyecto, Analista: seleccionar un caso de uso a editar</p> <p>50. Sistema: carga un caso de uso seleccionado</p> <p>51. Administrador, Jefe de Proyecto, Analista: ingresas nuevos datos para editar un caso de uso</p> <p>52. Sistema: caso de uso editado satisfactoriamente</p>
Flujo Alternativo	<p><i>El sistema no encuentra el registro.</i></p> <p>✓ Verificar que el registro exista.</p> <p>Administrador, Jefe de Proyecto, Analista no puede guardar el registro.</p> <p>✓ Verificar los datos del formulario, para que sean datos admitidos por éste o estén incompletos</p>
Frecuencia	Cada vez que se requiera editar un caso de uso

Tabla 33: Caso de Uso - Consultar Caso de Uso

Actores	✓ Administrador, Jefe de Proyecto, Analista, Cliente
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista, Cliente: escoge el proyecto al cual pertenece el caso de uso
Precondiciones	La necesidad debe estar creada.
Pos condición	Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar ver un caso de uso
Flujo normal	<p>9. Administrador, Jefe de Proyecto, Analista, Cliente: escoge el proyecto al que pertenece el caso de uso</p> <p>10. Sistema: cargar los datos de ese proyecto.</p> <p>11. Administrador, Jefe de Proyecto, Analista, Cliente: seleccionar ver caso de uso</p> <p>12. Sistema: muestra el registro del caso de uso</p>
Flujo Alternativo	<p><i>El sistema no encuentra el caso de uso.</i></p> <p>✓ Verificar que el caso de uso exista.</p>
Frecuencia	Cada vez que se requiera consultar un caso de uso

Tabla 34: Caso de Uso - Eliminar Caso de Uso

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: seleccionar el caso de uso que desea eliminar.
Precondiciones	El caso de uso debe estar registrado.
Pos condición	Administrador, Jefe de Proyecto, Analista: eliminar el caso de uso registrado
Flujo normal	<p>9. Administrador, Jefe de Proyecto, Analista: seleccionar el caso de uso que desea eliminar.</p> <p>10. Sistema: confirmar la eliminación del caso de uso</p> <p>11. Administrador, Jefe de Proyecto, Analista: elimina caso de uso</p> <p>12. Sistema: caso de uso eliminado satisfactoriamente.</p>
Flujo Alternativo	<p><i>El sistema no encuentra el caso de uso.</i></p> <p>✓ Verificar que el caso de uso exista.</p>
Frecuencia	Cada vez que se requiera eliminar un caso de uso

Tabla 35: Casos de Uso - Generar Informe y Estadística

Actores	✓ Administrador, Jefe de Proyecto, Analista
Personal involucrado e intereses	Administrador, Jefe de Proyecto, Analista: escoger estadística a generar.
Precondiciones	EL usuario debe estar registrado.
Pos condición	Administrador, Jefe de Proyecto, Analista: visualiza reporte generado.
Flujo normal	<p>1. Administrador, Jefe de Proyecto, Analista: selecciona el reporte a generar e ingresa datos para el reporte.</p> <p>2. Sistema: genera reporte.</p> <p>3. Administrador, Jefe de Proyecto, Analista: Visualiza reporte generado.</p>
Flujo Alternativo	<p><i>El sistema no genera el reporte.</i></p> <p>✓ Verifique datos de entrada del reporte.</p>
Frecuencia	Cada vez que se requiera visualizar un reporte.

6.4. RESULTADOS DEL OBJETIVO 4

Evaluar el producto haciendo pruebas unitarias y de laboratorios para determinar el grado de cumplimiento con los requisitos iniciales. El aseguramiento de calidad del software se compone de una serie de pruebas que atraviesan todo el ciclo de desarrollo e implementación de la aplicación. Realizando algunas pruebas como pruebas del sistema y las pruebas de funcionalidad

Se realizaron pruebas para medir el grado de cumplimiento del software en los requisitos iniciales, realizando pruebas y simuladas acorde a lo estipulado en el diseño y plan de prueba, *Anexo D – FORMATO DISEÑO Y PLAN DE PRUEBAS.*

Criterios de Evaluación de Diseño y Plan de Pruebas

Pruebas de Sistema

- Detectar errores en la ejecución de las pruebas
- Que los reportes generados por las herramientas de automatización de las pruebas contengan las mínimas variables que permitan un análisis acertado de cada una de las pruebas realizadas.
- Tener en cuenta todos los escenarios posibles.
- El 90% de las pruebas realizadas deben ser exitosas.

Pruebas Funcionales

- El resultado de cada caso de prueba debe ser igual al resultado de salida esperado.
- Encontrar fallas al ejecutar los diferentes casos de pruebas.
- La aplicación cumple con los requerimientos funcionales especificados en la fase de análisis.
- La aplicación cumple con los requerimientos mínimos para el funcionamiento.

Resultados de las Pruebas de Usabilidad

En esta fase de desarrollo se realizó la encuesta para pruebas de usabilidad, donde el personal

laboral de la empresa SOLINCES S.A.S., evaluaron el software para identificar y rectificar las deficiencias de usabilidad y erros existentes, con ello asegurar que el sistema sea fácil de aprender y de usar, su uso sea satisfactorio provea utilidad y cumpla las tareas para las cuales fue desarrollado.

Para realizarlo se programó una visita en las oficinas de SOLINCES S.A.S.; donde un total de 4 ingenieros con diferentes perfiles, interactuaron con el software asumiendo los diferentes módulos de este.

Las pruebas de usabilidad se guio por la siguiente estructura de encuesta:

Tabla 36: Estructura de Pruebas de Usabilidad

No	<i>Evalúe con una escala de 1 a 5, donde 1 es pésimo y 5 es excelente, el software en diferentes aspectos</i>	1	2	3	4	5
1	Facilidad de Aprendizaje: Indique qué tan fácil es aprender y manejar la funcionalidad básica del software.	<input type="checkbox"/>				
2	Suministro: Indique que tanto facilita el software la comunicación entre el cliente y las partes interesadas en la recolección de requisitos.	<input type="checkbox"/>				
3	Análisis de los Requisitos: Indique que tan fácil es registrar e identificar los requisitos del sistema y los requisitos del cliente.	<input type="checkbox"/>				
4	Planificación del proyecto: Indique con qué facilidad el software permite definir el alcance del proyecto y la asignación de tareas a las partes interesadas.	<input type="checkbox"/>				
5	Medición: Indique que tanto software permite desarrollar un conjunto de medidas a partir de dichas necesidades.	<input type="checkbox"/>				
6	Control del Proyecto: Indique que tan satisfecho esta con el reporte e informes sobre procesos del proyecto que genera el software.	<input type="checkbox"/>				
7	Gestión de la Configuración: Indique que tan satisfecho esta con interfaz diseñada para facilitar la realización eficiente de las tareas de la mejor forma posible	<input type="checkbox"/>				

Los resultados más relevantes obtenidos se encuentran en los gráficos estadísticos que se

presentaran adelante. Para observar cada una de las pruebas, diríjase a la carpeta de Anexos: Resultados diseño y plan de pruebas en donde encontrará las actas en digital.

Se desarrolló la encuesta de usabilidad dirigida al representante legal de SOLINCES S.A.S., con el manejo de los procesos. La encuesta se basa de preguntas claras que permiten validar el nivel de usabilidad del sistema.

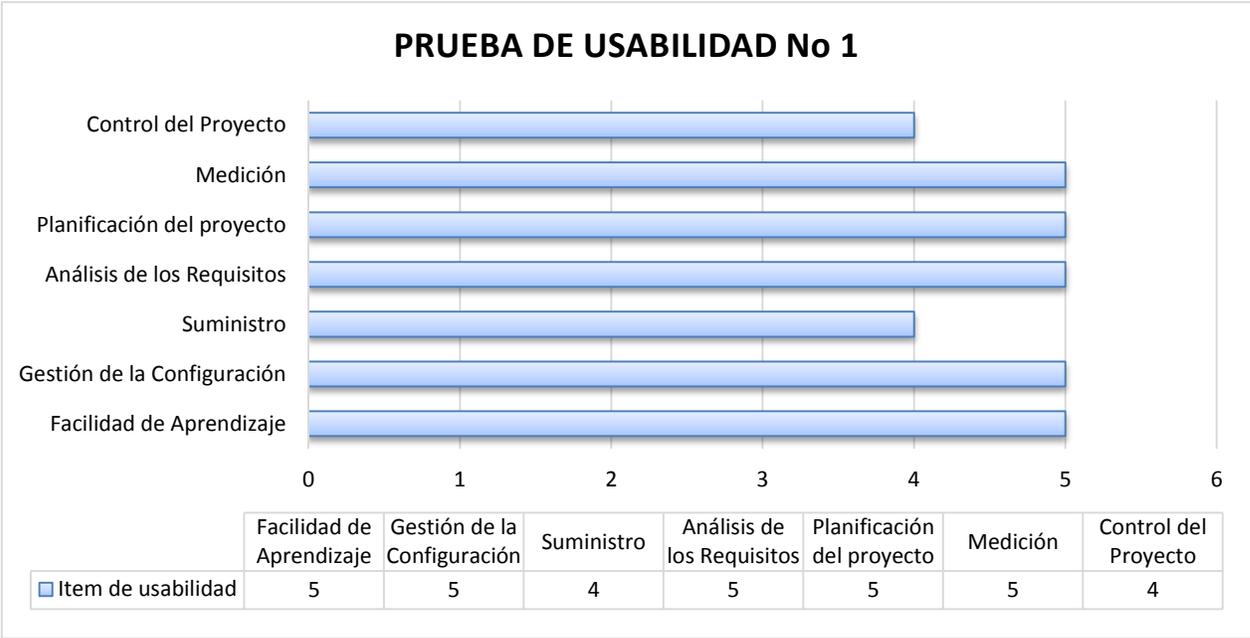


Ilustración 10: Encuesta de usabilidad realizada al representante legal. Fuente (Autores)

En la ilustración 10, se observa la encuesta de usabilidad dirigida al representante legal de SOLINCES S.A.S., esta encuesta arrojó resultados máximos en la prueba de medición, planificación del proyecto, análisis de los requisitos, gestión de la configuración y facilidad de aprendizaje, y una calificación alta en el control del proyecto y suministro con el número 4.

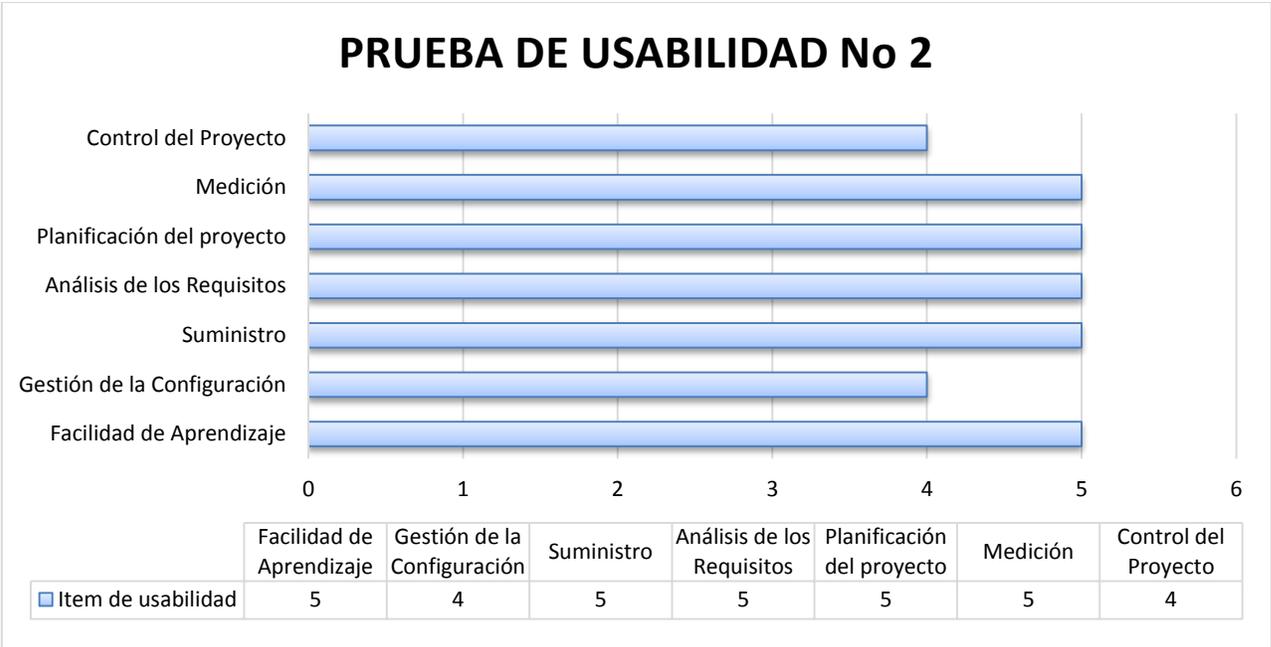


Ilustración 11: Encuesta de usabilidad realizada a la ingeniera jefe de proyectos. Fuente (Autores)

En la ilustración 11, se observa la encuesta de usabilidad dirigida a la ingeniera jefe de proyectos de SOLINCES S.A.S., esta encuesta arrojó resultados máximos en la prueba de medición, planificación del proyecto, análisis de los requisitos, facilidad de aprendizaje y suministro, y una calificación alta en el control del proyecto y gestión de la configuración con el número 4.

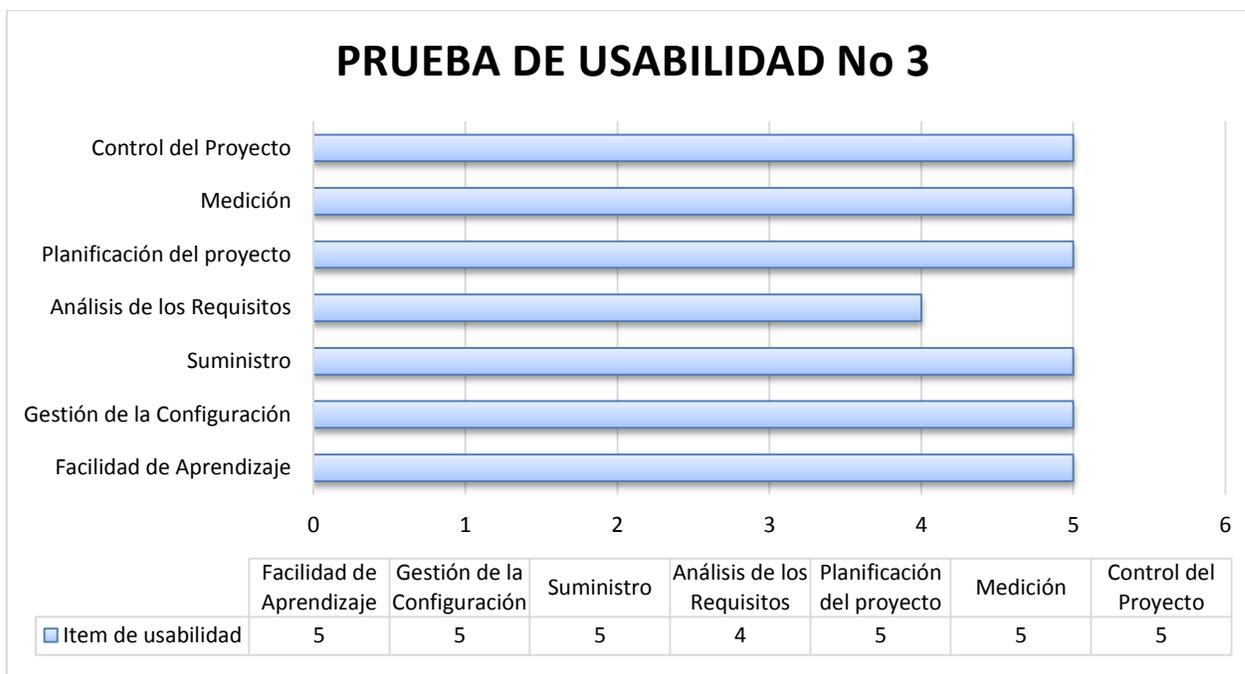


Ilustración 12: Encuesta de usabilidad realizada a los analistas. Fuente (Autores)

En la ilustración 12, se observa la encuesta de usabilidad dirigida a los analistas de SOLINCES S.A.S., esta encuesta arrojó resultados máximos en la prueba de medición, planificación del proyecto, facilidad de aprendizaje, control del proyecto y suministro, y una calificación alta en el y gestión de la configuración, análisis de los requisitos con el número 4.

Como se puede apreciar en las anteriores encuestas de usabilidad realizada al representante legal, ingeniera jefe de proyectos y a analistas en la empresa SOLINCES S.A.S., se aprecian valores satisfactorios en cada uno de los ítems, logrando casi un máximo valor en las pruebas. A lo que se refiere a la facilidad de aprendizaje, gestión de la configuración, planificación del proyecto, medición obtuvieron una calificación similar y la más alta propuesta por la prueba que es de un 5, y se puede concluir que es un sistema fácil en su aprendizaje y operación. La presentación visual, análisis de los requisitos tomaron valores que estuvieron entre 4 y 5 en la escala de calificación de la encuesta de usabilidad, lo que indica la satisfacción plena con el desempeño del sistema, y el cual cumple con todos los requerimientos funcionales.

Para observar los resultados de las pruebas de usabilidad, pruebas de sistema y pruebas de funcionalidad se encuentran en la carpeta de anexos F.

El presente proyecto en la ciudad de Cartagena de Indias no presenta antecedentes o implementaciones semejantes en otras empresas de desarrollo de software basándose en la norma ISO/IEC 15504 o modelo SPICE. Sin embargo, se espera que con el producto software como resultado de esta investigación, otras organizaciones interesadas en la implementación de procesos utilizando la norma ISO/IEC 15504 puedan aplicarlo de manera ágil, escalonada y efectiva permitiendo la escalabilidad a los próximos niveles del modelo SPICE.

En conclusión la prueba desarrollada en la empresa culminó satisfactoriamente con un alto grado de satisfacción de los funcionarios en SOLINCES S.A.S involucrados en este proyecto, evidencia de ello es el acta de aceptación del proyecto, firmada por el ingeniero Giraldo Turizo Cano, Representante legal, con fecha de 12 de septiembre de 2014. Ver anexo CARTA DE ACEPTACION DE SOFTWARE POR SOLINCES S.A.S.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

El objetivo principal de este proyecto hacía referencia a la interrelación del cliente con empresas desarrolladoras de software para la creación de un producto final esperado, para ello el software creado ha contribuido de manera muy importante para identificar los procesos de gestión de requisitos a través de la norma ISO/IEC 15504 (SPICE).

El proyecto ha logrado cumplir todos los objetivos básicos que se habían propuesto y respetando las especificaciones planteadas.

Al realizar el análisis de la norma ISO/IEC 15504 para la implementación en este proyecto se propuso a la empresa SOLINCES S.A.S. que la principal motivación para tener en cuenta un modelo de mejora de proceso es la necesidad de una mejora a nivel de gestión interna como la asignación de tareas, desarrollo de procesos, definición de etapas, entre otros, y la obligación de satisfacer en un mayor índice al cliente.

Se han desarrollado módulos para diferentes perfiles que obtienen los suministros, definición de requisitos de los stakeholders, análisis de los requisitos del sistema y planificación del proyecto a través de la asignación de tareas según el perfil. Todo esto se realiza basándose en los aspectos a tener en cuenta para implementar la norma ISO/IEC 15504 en los niveles 1 y 2 de madurez.

Para el diseño y desarrollo del sistema se utilizó como herramienta UML, que a través de sus diagramas se pudo crear la estructura del software del sistema. Con los modelos de caso de uso se representó la forma de cómo el usuario interactúa con el sistema y todas las operaciones que el usuario necesita que éste haga.

Las buenas prácticas de programación permitieron el desarrollo de un software con una interfaz amigable con el usuario lo cual satisface el requerimiento de ser apto para fines educativos; además que el resultado final fue un producto software que facilita la interrelación del cliente con los desarrolladores de software a través de los identificación de

las necesidades, requerimientos y funcionalidades para el producto final esperado por el usuario.

Al obtener el producto final se procedió a realizar las pruebas del sistema lo cual permitió observar la mejoría en los procesos en SOLINCES S.A.S., con ello la empresa sabe que las pruebas exitosas son la garantía que el sistema está bien, para la garantía del software es necesario probarlo, ya que va de por medio prestigio, fiabilidad, proyección, para proyectos futuros.

Con la creación del software la empresa SOLINCES S.A.S facilito la interacción en las actividades de gestión de requisito, recolección de la información, relación con los clientes, cumplimiento de calendarios y control de costos, generando mayor productividad y satisfacción por parte de los clientes en el producto final. Lo cual ayudaría a la organización a ser más competitiva y tener más acogida en la región

Existen muchas ventajas y beneficios para que SOLINCES S.A.S. siendo una empresas desarrolladoras de software pueda optar por la certificación en la norma ISO/IEC 15504, al ser ésta una forma de certificación más económica y capaz de adaptarse a las necesidades y a la estructura de trabajo definido por la empresa para el ciclo de vida de desarrollo de software.

El uso de la norma ISO/IEC 15504 (SPICE) en SOLINCES S.A.S facilita la integración con otras del sector de las tecnologías de la información y comunicación (sector TIC), tales como: ISO 27000 de seguridad, ISO 20000 de servicios de IT, ISO 9000 e ISO 12207 la cual es específica para la ingeniería de software.

Adicionalmente la ISO/IEC 15504, es una norma de carácter abierto, desarrollada por una de las entidades con gran reconocimiento mundial en cuanto a regulación y definición de estándares, otorga a la empresa que se certifique un mayor índice de competitividad internacional con otras empresas del área.

En algunos países latinoamericanos, como México, Brasil, entre otros, se ha adaptado la norma ISO/IEC 15504 para aplicarla al proceso de desarrollo de software y de certificación de las empresas, obteniendo grandes beneficios competitivos y haciendo que las mismas

logren entrar en el mercado de la industria de software mundial. Sin embargo, el proceso de certificación de una Pyme desarrolladora de software, implica un gran compromiso por parte de sus integrantes, ya que es una labor ardua y de equipo, que requiere una inversión económica y de tiempo, lo cual llevará a que los procesos de la organización se realicen de una forma más eficiente y organizada, ocasionando que tanto la organización como el cliente tengan una mayor satisfacción y reconocimiento del producto final.

Hasta el momento no se dispone de certificaciones internacionales (ISO) para la mejora de procesos a nivel de organización, siendo la publicación de ISO/IEC 15504 una oportunidad para que las organizaciones puedan obtener una certificación internacional. El modelo referente en la actualidad, CMMI, es un estándar de facto, de uso internacional pero no avalado por una organización internacional como ISO.

RECOMENDACIONES

El alcance de la norma es establecer un marco de referencia común para los procesos del ciclo de vida del software. Contiene procesos, actividades y tareas para aplicar durante el suministro, desarrollo, operación y mantenimiento de productos software basados en los niveles 1 y 2 de madurez. Los procesos son descritos en términos de lograr los propósitos y salidas.

Esta investigación se limitó a los aspectos relacionados con la gestión de requisitos, debido que cubrir otros procesos de la ingeniería de software se hace demasiado extenso y podrían abarcar niveles superiores al básico y el gestionado. Es apropiado saber que el nivel 1 de madurez consta de tres procesos, que son: suministro, definición de los requisitos de usuario y análisis de los requisitos del sistema, así el nivel 2 abarca los procesos de la gestión del modelo de ciclo de vida. Para los niveles superiores se requiere de periodos más extensos y estarían por fuera del alcance y tiempo del proyecto.

Establecer un plan de mantenimiento de la aplicación asegurando así la operatividad del software.

Planificar reuniones posteriores con el fin de establecer posibles mejoras al software.

Para trabajos futuros se recomienda abarcar el nivel 3 de madurez, el cual abarca las siguientes áreas de trabajo:

- Proceso de Análisis de Requisitos del Software
- Proceso de Diseño de la Arquitectura del Software
- Proceso de Diseño de la Arquitectura del Sistema
- Proceso de Gestión de Infraestructuras
- Proceso de Gestión de Recursos Humanos
- Proceso de Gestión de Riesgos
- Proceso de Gestión de la Decisión
- Proceso de Integración del Software
- Proceso de Integración del Sistema
- Proceso de Verificación del Software
- Proceso de Validación del Software

BIBLIOGRAFÍA

- ACIS. (2007). 5 ° ACIS Jornadas de Ingeniería del Software de Investigación, Gestión y Aplicaciones . *Aplicacion de procesos software*, (págs. 10-12). Bogotá.
- AENOR. (2009). Guía de implementación del modelo de procesos de calidad del desarrollo de software en el nivel 2 de madurez SPICE en las Pymes. *Prysm Equality*, 7-11.
- AENOR. (2012). *Organizaciones Certificadas en ISO 15504*. Recuperado el 16 de mayo de 2013, de <http://www.iso15504.es/>
- Alarcón Aldana, A. C., González Sanabria, J. S., & Rodríguez Torres, S. L. (2011). Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504. *Revista Virtual Universidad Católica del Norte*(34), 11-14. Obtenido de <http://revistavirtual.ucn.edu.co/index.php/RevistaUCN/article/view/339/652>
- Alarcón Aldana, A. C., González Sanabria, J. S., & Rodríguez Torres, S. L. (2011). Revista Virtual Universidad Católica del Norte. *Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504*(34), 11-14.
- ALARCON, A. (2011). Guía para Pymes desarrolladoras de software basado en la norma ISO/IEC 15504. *Revista Virtual Fundación Universitaria Católica del Norte No 34*.
- Asociación Española para la Calidad. (2013). Recuperado el 05 de mayo de 2013, de <http://www.aec.es/web/guest/centro-conocimiento/spice>
- Basurto, C. (2009). Implementación y evaluación de la NTP ISO/IEC 12207 in Evaluación de procesos con ISO/IEC 15504. *Revista Virtual Universidad Católica del Norte*.
- Cañon, M. (04 de octubre de 2011). *ING. DEL SOFTWARE: EMPRESAS COLOMBIANAS CERTIFICADAS EN CMMI*. Recuperado el 27 de mayo de 2013, de <http://mylecanon.blogspot.com/2011/10/empresas-colombianas-certificadas-en.html>
- CMMI Institute. (2010). *CMMI Institute - the home of Capability Maturity Model Integration*. Recuperado el 12 de Marzo de 2013, de <http://cmminstitute.com/>
- CSOFTMTY Concejo de software de Nuevo Leon. (18 de Noviembre de 2009). *Empresas con certificación CMMI*. Recuperado el 27 de mayo de 2013, de <http://www.csoftmty.org/node/225>
- Darwin Jimenez Garzon. (24 de octubre de 2007). *GESTIÓN DE PROYECTOS DE SOFTWARE*. Obtenido de <http://ingenieriadjg.blogspot.com/2007/10/gestin-de-proyectos-de-software.html>

- De Amescua, A., Lloréns, J., & García, Á. (2008). SPICE: UN MARCO PARA LA EVALUACIÓN DE PROCESOS SOFTWARE. *Universidad Carlos III de Madrid*, 3-11.
- de Amescua, A., Lloréns, J., & García, Á. (s.f.). *SPICE: UN MARCO PARA LA EVALUACIÓN DE PROCESOS SOFTWARE*. Obtenido de Departamento de Informática.Universidad Carlos III de Madrid
- EMI - Analisis de Sistemas II. (19 de Noviembre de 2013). *Metodologia RUP (Rational Unified Process)*. Obtenido de <http://www.slideshare.net/bernardolimachi/metodologia-rup-14288208>
- Fernández, C. (2012). Calidad y madurez en el desarrollo del software. *Dossier. Tecnologia de la Información*, 2.
- Florac, W., Park, R., & Carleton, A. (1997). Practical Software Measurement: Measuring for Process Management and Improvement. *USA: Software Engineering Institute.*, 1 - 12.
- Garzás, J. (28 de Febrero de 2010). *¿CMMI o ISO 15504 SPICE?* Recuperado el 12 de Febrero de 2013, de <http://www.javiergarzas.com>
- Garzás, J. (3 de Octubre de 2010). *Entender la norma ISO 15504 (1/3). Los aspectos básicos.* Recuperado el 03 de Agosto de 2012, de <http://www.javiergarzas.com>
- Garzás, J. (01 de Marzo de 2011). *Evaluacion y Mejoras de Procesos Software*. Recuperado el 14 de Marzo de 2013, de <http://albinogoncalves.files.wordpress.com/2011/03/jgarzas-evaluacionmejora1.pdf>
- Garzás, J., & Piattini, M. (2010). *Fábrica de software: Experiencias, tecnologías y organización*.
- Garzás, J., Fernández, C., & Piattini, M. (septiembre de 2009). *REICIS - Revista Española de Innovación, Calidad e Ingeniería del Software*, 5(2), 88-98. Obtenido de <http://www.ati.es/IMG/pdf/GarzasVol5Num2.pdf>
- Gómez, O., Oktaba, H., García, F., & Piattini, M. (2010). A systematic review in Software Engineering: State-of-the-art in measures. *Communications in Computer and Information Science*, 224 - 231. Obtenido de Communications in Computer and Information Science.
- Gonzalo, M.n. (19 de noviembre de 2012). *Bitelia*. Recuperado el 27 de mayo de 2013, de 10 herramientas de software libre para gestionar proyectos: <http://bitelia.com/2012/11/herramientas-software-libre-gestionar-proyectos>

- Huamatuco, D. H., Vargas, J. I., & Quispe, V. (2011). Metodología RUP (Rational Unified Process). *Universidad Nacional del Altiplano*.
- Instituto Nacional de Tecnologías de Comunicación. (2008). Estudio sobre la certificación de calidad como medio para impulsar el desarrollo de software en España. *INTECO Laboratorio nacional de calidad del software*, 27.
- Inteco. (2008). Estudio sobre la certificación de calidad como medio para impulsar el desarrollo de software en España. *Observatorio de la seguridad de la información*.
- INTECO. (Diciembre de 2008). *GUÍA PRÁCTICA DE* . Obtenido de INTECO. Instituto Nacional de Tecnologías y comunicación
- INTECO. (2009). *INTECO. Instituto Nacional de Tecnologías de Comunicación*. Recuperado el 03 de Agosto de 2012, de Instituto Nacional de Tecnologías de Comunicación. Ingeniería del software: Metodologías y ciclos de vida: INTECO. Instituto Nacional de Tecnologías de Comunicación
- Kendall, K., & Kendall, J. (2002). *Análisis y diseño de sistemas*. Estados Unidos: Prentice-hall hispanoamericana.
- Lee, E.-S., Kim, H.-k., & Hwang, S.-M. (2001). Análisis de la prioridad de los elementos de seguridad para el requisito de mejora de procesos en la norma ISO / IEC 15504 y 15408 ISO / IEC. *Elsevier*.
- LTDA, M. &. (2004). Panorama de la Industria. Latinoamericana de Software. 97.
- McCall, J., Richards, P., & Walters, G. (02 de Noviembre de 1977). *Factors in Software Quality. Volume I. Concepts and Definitions of Software Quality*. Recuperado el 12 de Marzo de 2013, de Factors in Software Quality. Volume I. Concepts and Definitions of Software Quality: <http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA049014>
- Piattini, M., & García, F. (2003). *Calidad en el desarrollo y mantenimiento del software*. RA-MA Editorial.
- Piattini, M., Pino, F., & Gracia, F. (2006). Adaptación de las Normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la Evaluación de la Madurez de Procesos Software en Países en Desarrollo. *IEEE Latin America Transactions*, vol. 4, N°. 2.
- Pino, F. J., García, F., Ruiz, F., & Piattini, M. (abril de 2006). Adaptación de las normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la evaluación de la madurez de procesos software en países en desarrollo. *IEEE América Latina*, 4(2), 85 - 92.

Obtenido de Revista IEEE América Latina:
http://issuu.com/santi1970/docs/4tla2_4pino.pdf

Pino, F. J., García, F., Serrano, M., & Piattini, M. (2006). Medidas para estimar el rendimiento y capacidad de los procesos software de conformidad con el estándar ISO/IEC 15504-5:2006. *Revista Española de Innovación, Calidad e Ingeniería del Software*, Vol 2(3), 17 - 30. Obtenido de <http://www.ati.es/IMG/pdf/PinoVol2Num3.pdf>

Pino, F. J., Serrano, M., Garcia, F., Piattini, M., & Oktaba, H. (2006). Medidas para Estimar el Rendimiento y Capacidad de los Procesos Software de Conformidad con ISO/IEC 15504 . *Competisoft*, 42.

PMI Project Management Institute. (2004). *Guia de los fundamentos de la direccion de proyectos*. Recuperado el 29 de mayo de 2013, de http://gio.uniovi.es/documentos/software/GUIA_PMBok.pdf

Popkin Software and Systems. (s.f.). *Modelado de Sistemas com UML*. Obtenido de <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/doc-modelado-sistemas-uml.pdf>

Pressman, R. (2010). *Ingeniería de software, un enfoque práctico. Sexta edición.* . España: McGraw-Hill Interamericana.

Puello Marrugo, P., & Garcia Ramirez, F. (27 de Noviembre de 2009). GESTIÓN DE REQUISITOS EN LA INGENIERÍA DE SOFTWARE. *Ingeniator*, 1(1), 71-80. Recuperado el 20 de noviembre de 2013, de <http://letravirtual.usbctg.edu.co/index.php/ingeniator/article/viewFile/137/155>

Ramirez Aguirre, P. A., & Ramirez Arias, C. (10 de Junio de 2010). *Respiratorio.utp.edu.co*. Recuperado el 25 de abril de 2013, de <http://repositorio.utp.edu.co/dspace/bitstream/11059/1977/1/0053R173e.pdf>

Rodríguez, P., Alonso, J., & Sánchez, J. (2005). ¿Cuál es la madurez que necesitarían los procesos para el desarrollo de sistemas de software crítico? *Revista Española de Innovación, Calidad e Ingeniería del software*, 1(2), 31-41. Obtenido de <http://redalyc.uaemex.mx/redalyc/pdf/922/92210204.pdf>

Rodriguez, P., Alonso, J., & Sanchez, J. C. (2005). *¿Cuál es la madurez que necesitarían los procesos para el desarrollo de sistemas de software crítico?* Recuperado el 27 de mayo de 2013, de Red de Revistas Científicas de América Latina y el Caribe, España y Portugal: <http://www.redalyc.org/articulo.oa?id=92210204>

- Sacha Moufarrege. (s.f.). *Herramientas usadas en la ingeniería del software*. Obtenido de http://www.ehow.com/list_6735194_tools-used-software-engineering.html
- Software, P. (16 de octubre de 2012). *ISO/IEC 15504:2004*. Obtenido de <http://procesossoftwareai.blogspot.com/2012/10/isoiec-155042004.html>
- Universidad Tecnológica de Pereira. (2009). MODELO DE CALIDAD DE SOFTWARE . *Scientia et Technica Año XV, No 42, 2*.
- Valencia, L., Villa, P., & Ocampo, C. (2009). Modelos de calida del software. *Scientia et Technica, Universida Tecnologica de Pereira, 2-4*.
- Vates S.A. (2013). *Vates S.A. - Ingeniería de Software ¿Que es CMMI? - Software Engineering Institute*. Recuperado el 26 de mayo de 2013, de <http://www.vates.com/cmmi/que-es-cmmi.html>
- Villalobos, J. (1993). *Análisis y Diseño Orientados por Objetos*. Bogotá: Universidad de los Andes.
- WordPress. (03 de octubre de 2010). *WordPress.com*. Recuperado el 02 de junio de 2013, de <http://misaelutec.wordpress.com/caracteristicas-de-los-modelos-cmmipsptsp-e-ieee-para-dante/>

